



**ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO**

**FACULTAD DE INFORMATICA Y ELECTRONICA**

**ESCUELA DE INGENIERIA EN SISTEMAS**

**“ESTUDIO COMPARATIVO DE LAS TECNOLOGIAS PYTHON Y PERL PARA DESARROLLAR APLICACIONES WEB IMPLEMENTADO AL PROGRAMA DE ALFABETIZACION DEL CONSEJO PROVINCIAL DE CHIMBORAZO”**

**TESIS DE GRADO**

Previa la obtención del Título de:

**INGENIEROS EN SISTEMAS INFORMATICOS**

**CARLOS PATRICIO CHAVEZ ÑAÑAY**

**GUSTAVO IVAN BUÑAY GUALOTO**

**RIOBAMBA – ECUADOR**

**2008**

Agradecemos a Dios y a nuestros padres por guiarnos día a día y permitimos realizar nuestros anhelos, a la Escuela Superior Politécnica de Chimborazo, institución que permitió nuestra formación, a sus autoridades y docentes en especial a nuestro tutor y miembro de nuestra tesis que guiaron nuestro camino para alcanzar la ansiada meta, al H.C.P.CH. por las facilidades para el desarrollo y culminación de la misma, a nuestros hermanos y amigos.

La realización de esta tesis la dedicamos con amor a nuestros padres, y hermanos que con su apoyo incondicional y sacrificio han permitido que culminemos una etapa más de nuestra vida.

Carlos Patricio Chávez Ñauñay.

Gustavo Iván Buñay Gualoto.

**FIRMAS RESPONSABLES Y NOTA**

DR. ROMEO RODRIGUEZ \_\_\_\_\_

**DECANO DE LA FACULTAD DE  
INFORMATICA Y ELECTRONICA**

ING. IVAN MENES \_\_\_\_\_

**DIRECTOR DE LA ESCUELA DE  
INGENIERIA EN SISTEMAS**

ING. DANILO PASTOR \_\_\_\_\_

**DIRECTOR TESIS**

ING. ROBERTO INSUATI \_\_\_\_\_

**MIEMBRO DE TESIS**

TLGO. CARLOS RODRIGUEZ \_\_\_\_\_

**DIRECTOR DEL CENTRO DE  
DOCUMENTACION**

**NOTA:** \_\_\_\_\_

## **RESPONSABILIDAD DE LOS AUTORES**

Nosotros, “Carlos Patricio Chávez Ñañañay, Gustavo Iván Buñay Gualoto, somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis, y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo”.

## INDICE DE ABREVIATURAS

<b>PYTHON</b>	Lenguaje orientado a objetos.
<b>E/S</b>	Entrada y Salida.
<b>GUI</b>	Interfase gráfica con el usuario.
<b>PERL</b>	Lenguaje de Programación.
<b>UNIX</b>	Sistema Operativo.
<b>WEB</b>	Navegador.
<b>APACHE</b>	Servidor de Base de Datos
<b>HTML</b>	HyperText Markup Language
<b>MYSQL</b>	Motor de Base de Datos
<b>SGBD</b>	Gestores de Base de Datos
<b>OSAL</b>	Capa abstracta del Sistema Operativo
<b>ODBC</b>	(Open-DataBase-Connectivity)
<b>HCPCH</b>	Honorable Consejo Provincial de Chimborazo
<b>OPEN SOURCE</b>	Código Abierto
<b>GPL</b>	Licencia de Python
<b>API</b>	Application Programming Interface
<b>CGI</b>	Common Gateway Interface (Interfaz externa Común).
<b>HTTP</b>	HiperText Markup Lenguaje. (Lenguaje de Marcado de Hipertexto).
<b>PSP</b>	PYTHON Server Pages
<b>ZOPE</b>	Z Object Publishing Environment

<b>ORB</b>	(Object Request Broker)
<b>JDBC</b>	Acceso a Base de Datos en Java
<b>DB-API</b>	API de Python
<b>CLR</b>	Common Language Runtime (Lenguaje Común en Tiempo de Ejecución).
<b>MS-DOS</b>	Microsoft Disk Operating System (Sistema Operativo de Disco de Microsoft).
<b>ESPOCH</b>	Escuela Superior Politécnica de Chimborazo.
<b>FTP</b>	File Transfer Protocol. (Protocolo de Transferencia de Archivos).
<b>GNU</b>	Es un acrónimo recursivo que significa "GNU No es Unix".
<b>GPL</b>	General Public License (Licencia General Pública).
<b>GTK</b>	Gimp Tool Kit.
<b>IP</b>	Protocol Internet (Protocolo de Internet).
<b>ISO</b>	Sistema Internacional para la Estandarización.
<b>PHP</b>	Personal Hypertext Preprocessor (Personal Hipertext Procesor).
<b>RDBMS</b>	Sistema de Manejo de base de Datos Relacional.
<b>SEPROCH</b>	Seguimiento y Evaluación de Proyectos en Chimborazo.
<b>SQL</b>	Language Query Structure (Lenguaje de Consultas Estructurado).
<b>URL</b>	Uniform Resource Locator (Localizador Uniforme de Recurso).
<b>SISALFA</b>	Sistema de Alfabetización
<b>DSDM</b>	Dynamic Systems Development Method

## INDICE GENERAL

**PORTADA**

**AGRADECIMIENTO**

**DEDICATORIA**

**ÍNDICE DE ABREVIATURAS**

**ÍNDICE GENERAL**

**ÍNDICE DE TABLAS**

**ÍNDICE DE FIGURAS**

**INTRODUCCIÓN**

### CAPITULO I

Marco Referencial.....	- 18 -
1.1.1 Servidor Web.....	- 20 -
1.1.2 Arquitectura y tipos servidores Web.....	- 20 -
1.1.3 Servidor Apache .....	- 22 -
1.1.4 Gestor de Base de Datos MySql .....	- 23 -
1.1.4.1 Características: .....	- 23 -
1.1.4.2 MySql Control Center. ....	- 24 -
1.2 Problematización.....	- 24 -
1.3 Justificación.....	- 26 -
1.4 Objetivos.....	- 28 -
1.4.1 Objetivo General.....	- 28 -
1.4.2 Objetivos Específicos.....	- 28 -
1.5 Hipótesis .....	- 29 -

### CAPITULO II

#### ASPECTOS GENERALES DE LAS TECNOLOGIAS PYTHON Y PERL

2.1 Lenguaje PYTHON.....	- 30 -
2.1.1 Introducción a PYTHON .....	- 30 -
2.1.2 Características de PYTHON.....	- 33 -
2.1.3 Para que no es útil y si es útil PYTHON.....	- 34 -
2.1.4 Programación Web en PYTHON .....	- 35 -
2.1.4.1 Servidores de aplicaciones.....	- 36 -
2.1.4.2 Integración Apache/PYTHON: mod_python .....	- 37 -
2.1.4.2 Configuración Apache y Mod_Python.....	- 38 -
2.1.4.3 Instalando Apache y mod_python.....	- 39 -
2.1.4.4 Python Server Pages en mod_python.....	- 39 -



2.1.4.5 Sintaxis .....	41 -
2.1.4.5 Variables.....	43 -
2.1.4.6 Tipos de datos .....	43 -
2.2 Lenguaje PERL .....	44 -
2.2.1 Introducción a PERL.....	44 -
2.2.2 Características de PERL.....	45 -
2.2.3 Programación Web con PERL.....	47 -
2.2.4 CGIs de Apache.....	47 -
2.2.5 Tipos de Datos en PERL.....	51 -
2.2.6 Sintaxis en PERL .....	55 -
2.2.7 Sentencias de Control.....	56 -
2.3 PERL vs PYTHON.....	58 -
2.3.1 Ventajas y Desventajas de PERL y PYTHON .....	58 -
2.3.1.1 Ventajas de PERL .....	58 -
2.3.1.2 Desventajas de PERL .....	59 -
2.3.1.3 Ventajas de PYTHON .....	60 -
2.3.1.4 Desventajas de PYTHON .....	60 -
2.3.2 Semejanzas y diferencias entre PYTHON y PERL .....	61 -
2.3.2.1 Semejanzas entre PERL y PYTHON .....	61 -
2.3.2.2 Diferencias.....	62 -
2.3.2.3 Fortalezas y Debilidades de Perl y Python.....	62 -

### CAPITULO III

#### ANÁLISIS COMPARATIVO DE LAS TECNOLOGIAS PERL Y PYTHON

3.1 Introducción .....	64 -
3.2 Determinación de las tecnologías a comparar.....	65 -
3.3 Análisis de las Tecnologías Seleccionadas .....	65 -
3.3.1 PERL.....	65 -
3.3.1.1 Manipulación de base de Datos con PERL .....	66 -
3.3.1.2 Preparando nuestra Base de Datos .....	68 -
3.3.2 PYTHON.....	70 -
3.3.2.1 Manipulación de Base de Datos con PYTHON.....	70 -
3.3.3 Instalación y Configuración de Perl y Python en Linux Centos 4.3.....	71 -
3.3.3.1 Configuración de los Módulos e PERL y PYTHON en Apache.....	73 -
3.4 Determinación de los Parámetros de Comparación.....	76 -
3.4.1 Acceso a la Base de Datos.....	76 -
3.4.2 Líneas de Código .....	76 -
3.4.5 Interfaz .....	77 -
3.5 Descripción de los Módulos de Prueba .....	77 -
3.5.1 Modulo 1 .....	77 -
3.5.2 Módulo 2 .....	77 -
3.5.3 Módulo 3 .....	78 -
3.5.4 Módulo 4 .....	78 -
3.6 Desarrollo de los Módulos de Prueba.....	78 -
3.6.1 Modulo 1 .....	78 -
3.6.1.1 Modulo1 en Python.....	78 -
3.6.1.2 Módulo1 en Perl.....	80 -

3.6.2 Modulo 2 .....	- 82 -
3.6.2.1 Modulo 2 en Python. ....	- 82 -
3.6.2.1 Modulo 2 en Perl. ....	- 86 -
3.6.3 Módulo 3 .....	- 89 -
3.6.3.1 Modulo 3 en Python. ....	- 89 -
3.6.3.2 Modulo 3 en Perl. ....	- 90 -
3.6.4 Módulo 4 .....	- 91 -
3.6.4.1 Modulo 4 en Python. ....	- 91 -
3.6.4.2 Módulo 4 en Perl. ....	- 92 -
3.7 Análisis Comparativo .....	- 93 -
3.7.1 Criterios de Comparación.....	- 93 -
3.7.2 Análisis de los Criterios de Comparación. ....	- 98 -
3.7.2.1 Acceso a Base de Datos.....	- 98 -
3.7.2.2 Líneas de Código.....	- 102 -
3.7.2.3 Portabilidad.....	- 105 -
3.7.2.4 Interfaz de Usuario .....	- 108 -
3.7.3. Puntajes Alcanzados .....	- 110 -
3.7.4 Interpretación Final de Resultados.....	- 115 -

## CAPITULO IV

### DESARROLLO DEL SISTEMA SISALFA

Fase 1: PREPROYECTO .....	- 117 -
4.1 Identificación del Proyecto .....	- 117 -
4.1.1 Financiamiento y Comisión del proyecto.....	- 117 -
4.1.2 Reglas.....	- 117 -
Fase 2: ETAPAS VITALES DEL PROYECTO.....	- 119 -
4.2 Etapa 1: El Estudio De Viabilidad .....	- 119 -
4.2.1 Requisitos previos para usar DSDM.....	- 119 -
4.2.2 INGENIERIA DE LA INFORMACIÓN.....	- 120 -
4.2.2.1 Definición del Ámbito.....	- 120 -
4.2.2.2 Requerimientos .....	- 121 -
4.2.2.3 Estudio de Factibilidad .....	- 121 -
4.2.2.4 Planteamiento de la solución .....	- 122 -
4.2.2.5 Planificación Temporal .....	- 123 -
4.2.2.6 Especificación de Requerimientos (SRS).....	- 123 -
4.3 Etapa 2: Analisis del Sistema. ....	- 124 -
4.3.1 Casos de Uso del Sistema.....	- 124 -
4.3.1.1 Casos de Usos de Alto nivel Super_Administrador .....	- 124 -
4.3.1.2 Casos de Uso de alto nivel Administrador .....	- 124 -
4.3.1.3 Casos de Uso de Alto nivel Usuario.....	- 125 -
4.3 Etapa 3: Iteración Modelo Funcional .....	- 125 -
4.3.1 Análisis.....	- 126 -
4.3.1.1 Definir y Refinar los casos de uso .....	- 126 -
4.4 Etapa 4. DISEÑO .....	- 130 -
Definición de Informes e Interfaces de Usuario. ....	- 130 -
4.4.1 Definición de la información de la interfaz de usuario. ....	- 130 -
4.4.2 Lenguaje de comunicación. ....	- 130 -

4.4.3 Diagramas de Interacción .....	- 131 -
4.4.3.1 Diagramas de Secuencia .....	- 131 -
4.4.3.2 Diagramas de Colaboración.....	- 132 -
4.4.3.3 Diagramas de Calles .....	- 133 -
4.4.3.4 Diagramas de Base de Datos .....	- 134 -
4.4.3.5 Diagramas de Despliegue .....	- 134 -
Diagrama de Componentes .....	- 135 -
Diagrama de Nodos .....	- 135 -
4.5 Etapa 5.- Implementación y Pruebas .....	- 136 -

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMARY

GLOSARIO

ANEXOS

## INDICE DE TABLAS

### CAPITULO II

Tabla N° II. 1 Versiones de Python .....	- 32 -
Tabla N° II. 2 Tipos de Datos en Python .....	- 43 -

### CAPITULO III

Tabla N°III. 1 Escala de Valoraciones .....	- 96 -
Tabla N°III. 2 Escala de Valoración Cualitativa.....	- 96 -
Tabla N°III. 3 Pesos para los Parámetros .....	- 97 -
Tabla N°III. 4 Porcentajes de los Resultados.....	- 98 -
Tabla N°III. 5 Calificación de Tecnologías .....	- 98 -
Tabla N°III. 6 Acceso a Base de Datos Tecnología Perl .....	- 99 -
Tabla N°III. 7 Acceso a Base de Datos Tecnología Python .....	- 99 -
Tabla N° III. 8 Resultados Parámetros Acceso a Base de Datos .....	- 100 -
Tabla N°III. 9 Líneas de Código Tecnología Perl.....	- 102 -
Tabla N°III. 10 Líneas de Código tecnología Python .....	- 102 -
Tabla N°III. 11 Resultados Parámetro Líneas de Código.....	- 103 -
Tabla N°III. 12 Parámetro Potabilidad Tecnología PERL.....	- 105 -
Tabla N°III. 13 Parámetro Portabilidad Tecnología PYTHON .....	- 105 -
Tabla N°III. 14 Resultado Parámetro Portabilidad.....	- 106 -
Tabla N°III. 15 Parámetro Interfaz Tecnología PERL .....	- 108 -
Tabla N°III. 16 Parámetro Interfaz Tecnología PYTHON .....	- 108 -
Tabla N°III. 17 Resultado Parámetro Interfaz .....	- 109 -
Tabla N°III. 18 Tabla General de Resultados .....	- 111 -

### CAPITULO IV

TablaN°IV 1 Tabla Típica deEventos.....	- 127 -
TablaN°IV 2 Curso típico de Eventos Administrador.....	- 128 -
TablaN°IV 3 Curso Típico de Eventos usuario.....	- 129 -

## INDICE DE FIGURAS

### CAPITULO II

Figura N° II. 1 Mysql en Linux.....	- 68 -
Figura N° II. 2 Tablas de la Base de Datos.....	- 69 -

### CAPITULO III

Figura N° III. 1 Mysql en Linux.....	- 68 -
Figura N° III. 2 Tablas de la Base de Datos.....	- 69 -
Figura N° III. 3 Configuración de Módulos Perl y Python en Linux .....	- 73 -
Figura N° III. 4 Módulos de Python en Linux .....	- 74 -
Figura N° III. 5 Asignación de Sitios Virtuales .....	- 74 -
Figura N° III. 6 Ubicación de los Módulos Perl y Python.....	- 75 -
Figura N° III. 7 Verificación de Módulos Cargados .....	- 75 -
Figura N° III. 8 Módulos Perl y Python.....	- 76 -
Figura N° III. 9 Base de Datos en Mysql.....	- 79 -
Figura N° III. 10 Pagina .psp .....	- 79 -
Figura N° III. 11 Pagina .psp .....	- 80 -
Figura N° III. 12 Ejecución Modulo 1 Python.....	- 80 -
Figura N° III. 13 Base de Datos para acceder por medio de Perl.....	- 81 -
Figura N° III. 14 Pagina .pl para acceder a la Base de Datos .....	- 81 -
Figura N° III. 15 Pagina Index.pl .....	- 82 -
Figura N° III. 16 Datos Ingresados en la Base de Datos. ....	- 82 -
Figura N° III. 17 Base de Datos para el Modulo 2.....	- 83 -
Figura N° III. 18 Pagina .psp .....	- 83 -
Figura N° III. 19 Formulario Ingreso de Datos Cantón.....	- 84 -
Figura N° III. 20 Datos Ingresados.....	- 84 -
Figura N° III. 21 Datos Ingresados en la Base de Datos .....	- 84 -
Figura N° III. 22 Formulario para la Actualización de Datos.....	- 85 -
Figura N° III. 23 Datos Ingresados para la Actualización de Datos .....	- 85 -
Figura N° III. 24 Formulario para la Eliminación de Datos .....	- 86 -
Figura N° III. 25 Campos de la Base de Datos .....	- 86 -
Figura N° III. 26 Paginas .pl .....	- 87 -
Figura N° III. 27 Formulario para Ingresar Datos en Perl .....	- 87 -
Figura N° III. 28 Datos a ser Insertados .....	- 88 -
Figura N° III. 29 Datos Ingresados en la Base de Datos .....	- 88 -
Figura N° III. 30 Link para Actualizar Datos .....	- 88 -
Figura N° III. 31 Datos a ser Actualizados .....	- 89 -
Figura N° III. 32 Link para Eliminar Datos .....	- 89 -
Figura N° III. 33 Modulo Python Instalado en Linux .....	- 90 -
Figura N° III. 34 Acceso a Base de Datos en Python por medio de Consola.....	- 90 -
Figura N° III. 35 Módulos Perl en Linux.....	- 91 -
Figura N° III. 36 Interfaz de para la programación en Python.....	- 92 -
Figura N° III. 37 Programación en Python .....	- 92 -
Figura N° III. 38 Interfaz Perl Express .....	- 93 -

Figura N° III. 39 Interfaz Perl Express .....	- 93 -
Figura N° III. 40 Grafico resultado del Parámetro Acceso a Base de Datos .....	- 101 -
Figura N° III. 41 Grafico resultado del Parámetro Líneas de Código .....	- 104 -
Figura N° III. 42 Grafico de Resultado de Portabilidad .....	- 107 -
Figura N° III. 43 Gráfico Resultado Parámetro Interfaz .....	- 110 -
Figura N° III. 44 Gráfico General de Resultados .....	- 112 -
Figura N° III. 45 Gráfico Resultado Final .....	- 113 -

## **CAPITULO IV**

FiguraN° IV 1 Etapas Vitales del Proyecto .....	- 119 -
FiguraN° IV 2 Modelo Funcional.....	- 125 -
FiguraN° IV 3 Grafico Caso de Uso SuperAdministrador .....	- 127 -
FiguraN° IV 4 Grafico Caso de uso Administrador .....	- 129 -
FiguraN° IV 5 Grafico Caso de Uso Usuario.....	- 130 -
FiguraN° IV 6 Nivel Arquitectonico .....	- 131 -
FiguraN° IV 7 Diagrama de secuencia Verificación de usuarios logueados .....	- 131 -
FiguraN° IV 8 Diagrama de secuencia Administrador.....	- 131 -
FiguraN° IV 9 Diagrama de secuencia Usuario .....	- 132 -
FiguraN° IV 10 Diagrama de colaboración Verificación de usuarios logueados ....	- 132 -
FiguraN° IV 11 Diagrama de colaboración Administrador .....	- 132 -
FiguraN° IV 12 Diagrama de colaboración Verificación de servicios .....	- 133 -
FiguraN° IV 13 Diagrama de calles .....	- 133 -
FiguraN° IV 14 Diagrama de base de datos.....	- 134 -
FiguraN° IV 15 Diagrama de componentes .....	- 135 -
FiguraN° IV 16 Diagrama de nodos .....	- 135 -

## **INTRODUCCION**

En los últimos tiempos, Internet ha tenido un auge espectacular. En concreto, la tecnología Web ha demostrado ser extremadamente simple y flexible para publicar y encontrar información. Sin embargo, las potencialidades de algunas de las tecnologías de programación Web no han sido aprovechadas en su totalidad hasta la actualidad por lo que surge el interés creciente de entre los desarrolladores en conocer los beneficios y bondades de la utilización de una u otra tecnología.

Dentro del proceso de desarrollo de una aplicación Web existen parámetros que deben ser tomados en cuenta, tales como: Acceso a Base de Datos, Líneas de código para operaciones fundamentales, portabilidad, e Interfaz de desarrollo. Todos estos parámetros son importantes para presentar al usuario final un producto que sea de su completo agrado y fácil de manejar.

Por todo lo señalado, surge el interés del estudio comparativo de tecnologías de código libre para desarrollo Web, que ayudará a obtener la tecnología que proporcione los mejores beneficios para realizar una aplicación Web.

El consejo Provincial de Chimborazo, a través del departamento de Gestión Social lleva a efecto la ejecución del programa de Alfabetización “Yo Si Puedo” contando con la participación de asesores, educadores, y coordinadores de la provincia como también con delegados de la República de Cuba. Por la política actual del Gobierno Nacional surge la necesidad de crear una aplicación Web para el programa de alfabetización en un lenguaje de código libre para facilitar su mantenimiento.

Para cumplir con lo anterior se realizó un estudio de los Lenguajes Perl y Python para el desarrollo de aplicaciones Web, ya que estos lenguajes son OpenSource y han sido difundidos hace algún tiempo atrás, encontrándose en la actualidad en sitios importantes de preferencia entre los desarrolladores por su capacidad y potencialidad en el desarrollo de aplicaciones.

Todas las ventajas, desventajas, configuraciones y potencialidades de los lenguajes así como también parte de su sintaxis y tipos variables las describimos en el capítulo II como parte de nuestro estudio.

Bajo estas condiciones, nuestro estudio comparativo, desarrollado en el capítulo III, tiene por objetivo demostrar en forma práctica (a través de módulos de prueba) y de forma teórica (a través de investigaciones y puntajes obtenidos) los beneficios, fortalezas, debilidades, facilidades, etc. de las tecnologías (Perl y Python) en el



desarrollo y desenvolvimiento de cada parámetro de comparación, y de cada una de sus variables expuestas en este capítulo.

El trabajo realizado servirá de soporte en el futuro para la toma de decisiones de cual de las dos tecnologías mencionadas anteriormente es la más factible para desarrollar aplicaciones Web.

Con la tecnología seleccionada en el capítulo III se procederá a desarrollar una aplicación Web para el programa de alfabetización del Consejo Provincial. Todo lo referente al hardware y Software utilizado así como también sobre la factibilidad y análisis del problema encontrado se detalla en el capítulo IV.

## **CAPITULO I**

### **Marco Referencial**

#### **1.1 Antecedentes**

Dada la popularidad de Internet, el desarrollo de aplicaciones con interfaz Web, independiente de la plataforma donde se ejecute el cliente Web (navegador), ha experimentado un crecimiento exponencial. A pesar del menor desarrollo de las interfaces Web respecto a las interfaces gráficas de aplicaciones, su facilidad tanto de uso, dada la familiaridad de los usuarios con el paradigma Web, como su simplicidad de desarrollo, son razones suficientes para considerarlas como primera opción en cualquier desarrollo que una empresa hoy en día.

En las aplicaciones Web, una sola instancia ejecutándose en un servidor Web (o varias en un cluster de servidores), es compartida por varios (en algunos casos cientos o miles) de usuarios que acceden a ella desde un cliente universal, o sea desde el navegador de Internet. El principal cometido de una aplicación Web es crear páginas HTML (o en cualquier otro lenguaje de marcado) de manera dinámica. Es decir, los datos estáticos predefinidos de una página son combinados con datos generados por la lógica de la

aplicación u obtenidos de fuentes de datos en tiempo real, por ejemplo de una base de datos.

Durante la ejecución concurrente de las tareas requeridas simultáneamente por los diferentes usuarios de la aplicación Web es frecuente que se detecten conflictos de acceso y modificación a datos en el servidor. Además, la aplicación Web debe generar las páginas Web, ejecutando toda la lógica de negocio subyacente, de la manera más rápida y eficiente posible. Es por tanto necesario, adoptar mecanismos en la parte servidora que agilicen los accesos a datos y soporten transacciones, es decir, conjuntos de operaciones físicas relacionadas sobre datos, que aparecen al cliente como una unidad lógica. Entran en juego aquí los llamados sistemas gestores de bases de datos (SGBD). La versión más común utilizada de los mismos son los sistemas relaciones de bases de datos. En definitiva, servidores Web y bases de datos relacionales conforman un binomio esencial en la creación de toda aplicación Web.

Una gran variedad de Empresas han visto la necesidad de publicar sus servicios en el Web, facilitando de esta manera al cliente realizar sus transacciones desde cualquier computador conectado al Internet.

Detrás de la aplicación encargada de presentar la información al usuario existen programas diseñados para procesar datos que son enviados al servidor.

PERL y PYTHON son alternativas muy atractivas porque son lenguajes especializados en buscar, extraer y presentar información desde un servidor de base de datos.

En la actualidad estos lenguajes han evolucionado de forma que es posible realizar labores de administración en cualquier sistema operativo. Esto se debe en gran parte a

su popularidad, a sus intérpretes que se distribuyen de forma gratuita. Para la desarrollar aplicaciones Web sea en PERL o en PYTHON se necesita además el servidor (que en este caso es Apache) y el motor de la base de datos MySQL.

### **1.1.1 Servidor Web**

Un servidor Web es un programa que sirve datos en forma de páginas Web, hipertextos o páginas HTML (HyperText Markup Language), textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.

La comunicación de estos datos entre cliente y servidor se hace por medio un protocolo el cual define el conjunto de reglas que gobiernan el intercambio entre entidades dentro de una red; concretamente del protocolo HTTP (Hyper Text Transfer Protocol) el cual carece de estado; es decir, cada petición de un cliente a un servidor no es influida por las transacciones anteriores, por lo que el servidor trata cada petición como una operación totalmente independiente del resto. Además una de las características del protocolo HTTP es que no es permanente, es decir, cada operación HTTP implica una conexión con el servidor, que es liberada al término de la misma.

### **1.1.2 Arquitectura y tipos servidores Web**

La arquitectura de un servidor Web, generalmente está dividida en la capa de servidor y en la capa soporte, las cuales contienen los siguientes subsistemas:

- Capa servidor: Esta capa contiene cinco subsistemas, que son los responsables de implementar la funcionalidad de un servidor Web.

- Subsistema de recepción: representa la primera “línea de ataque” y su labor consiste en esperar las peticiones HTTP de los clientes que llegan por la red. También, analiza las peticiones y determina las capacidades de los navegadores (tipo de navegador, compatibilidad, etc.). Este subsistema contiene la lógica necesaria para manejar múltiples peticiones.
- Analizador de peticiones: encargado de traducir la localización del recurso de la red al nombre del archivo local. Por ejemplo, la solicitud del recurso `http://www.mec.es` se traduce al fichero local `/var/www/webfiles/index.html`.
- Control de acceso: sirve para autenticar y permitir el acceso.
- Manejador de recursos: Este subsistema es el responsable de determinar el tipo de recurso solicitado; lo ejecuta y genera la respuesta.
- Registro de transacción: se encarga de registrar todas las peticiones y su resultado.
- Capa soporte: Esta capa actúa como una interfase entre el sistema operativo y el servidor Web y, entre los propios subsistemas de la capa superior.
- Útil: contiene funciones que son utilizadas por el resto de subsistemas.
- Capa abstracta del Sistema Operativo (OSAL): este subsistema encapsula el funcionamiento específico del sistema operativo para facilitar la portabilidad del servidor Web a diferentes plataformas.

Entre los tipos de servidores Web se encuentran los basados en procesos los cuales son el predecesor de todos los demás y que se basan en la obtención de paralelismo mediante la duplicación del proceso de ejecución, de donde surgieron los basados en hilos en los cuales se tiene la ventaja que la creación de un hilo no es tan costosa como

de un proceso. Además se encuentran los servidores basados en sockets no bloqueantes o dirigidos por eventos los cuales utilizan una llamada al sistema para examinar el estado de los 8 sockets con los que trabaja. Por último se tienen los servidores implementados en el núcleo con lo que se busca acelerar la velocidad de un servidor Web mediante el movimiento de su código de espacio de usuario a espacio de núcleo o kernel.

### **1.1.3 Servidor Apache**

El paquete Apache contiene un servidor HTTP de código abierto. Es útil para crear sitios Web en la intranet local o sirviendo enormes operaciones Web.

La Fundación Apache publica una actualización del servidor Web Apache 2.x, que elimina dos vulnerabilidades de seguridad que pueden ser utilizadas en ataques de denegación de servicio.

Apache es el software para servidores Web más popular existente en la actualidad. Según las estadísticas de Netcraft, en Mayo de 2003 existen más de 25 millones de servidores Web que utilizan Apache accesibles en Internet, lo que representa más del 60% del total de servidores Web en el mundo.

En la actualidad existe el servidor Web Apache 2.0.x. Esta última es una importante actualización de las versiones 1.3, pensada para ofrecer algunas prestaciones nuevas (como la utilización de threads, de forma que se mejora la capacidad de escalabilidad), una mayor compatibilidad en entornos no Unix (Windows, OS/2, BeOS), soporte de Ipv6, etc. Por otra parte, Apache 2.0.x viene a solucionar algunas de las principales

limitaciones o situaciones problemáticas de Apache 1.3.x, como pueden ser los problemas de prioridad de los módulos.

Acaba de publicarse una nueva versión de la "rama" 2.0 que tiene por objetivo solucionar dos vulnerabilidades de seguridad que pueden ser utilizadas en ataques de denegación del servicio. La primera de las nuevas vulnerabilidades, que afecta a las versiones 2.0.37 y superiores hasta la 2.0.45, provoca la detención inesperada del servidor y puede ser utilizada por un atacante remoto. No se han publicado los detalles específicos de esta vulnerabilidad, que se harán públicos el 30 de Mayo.

La segunda nueva vulnerabilidad afecta a las versiones 2.0.40 y posteriores hasta la 2.0.45 en las plataformas Unix. Se trata de un problema en el módulo de autenticación básica y puede ser utilizado por un atacante remoto para provocar una condición de denegación de servicio, impidiendo la validación de los usuarios hasta que Apache no sea reinicio.

#### **1.1.4 Gestor de Base de Datos MySql**

MySql es el más popular gestor de base de datos de código libre, es desarrollado, es distribuido y sustentado por MySql AB. MySql AB es una compañía comercial fundada por los desarrolladores que proporcionan servicios alrededor de la base de datos de MySql. Es una base de datos que recopilación de datos estructurados. Como por ejemplo puede ser algo de una lista de tiendas, una simple galería de cuadros o también una gran cantidad de información en una red corporativa.

##### **1.1.4.1 Características:**

- **Portable.-** Facilidad de transportación.
- **Variedad de tipos de datos como.-** integer, char, varchar, float, double, etc.
- **Comandos y funciones.-** Soporta operadores y funciones en el select y en el where.
- **Seguridad.-** Es muy flexible y seguro ya que es un sistema que permite la autenticación. La autenticación es segura porque todo el tráfico de claves son encriptadas cuando usted se conecta al servidor.
- **Conectividad.-** Los clientes pueden conectarse al servidor de MySql usando el protocolo TCP/IP Sockets, Unix Sockets y todo lo que tenga que ver con la tecnología NT. Además la conexión puede realizarse por medio de ODBC (Open-DataBase-Connectivity) con soporte para Win32.
- **Clientes y herramientas.-** Incluye myisamchk, una forma muy fácil de chequear las tablas, optimización y reparación. Todas estas funciones se realizan también a través de una interfaz SQL.

#### **1.1.4.2 MySql Control Center.**

Es una interfaz gráfica entre el gestor de base de datos MySql y la aplicación Web, hace uso de una conexión al servidor de base de datos. Además permite el uso de sentencias sql para creación de tablas, inserciones, actualizaciones y eliminaciones.

#### **1.2 Problematización.**

El departamento de Gestión Social del Consejo Provincial de Chimborazo, en la actualidad lleva a efecto el programa de Alfabetización en forma manual, es decir que los datos de los alfabetizados como de los encargados de cada programa son registrados en forma manual, e ingresados a hojas electrónicas en el mejor de los casos. Todo esto



lleva a que los usuarios tengan dificultades en muchos aspectos puesto que si necesitan un informe de cómo marcha el programa de alfabetización en un determinado cantón, parroquia o comunidad, esta información no puede ser entregada al instante sino luego de un tiempo.

La ejecución del programa empieza con la entrega de la solicitud en el departamento de Gestión Social del H. Consejo Provincial solicitando el inicio del programa de Alfabetización en una determinada comunidad, esta solicitud es aceptada para luego realizar una inspección en la comunidad para ver los requerimientos básicos para el inicio del programa. Dentro de esto se encuentra el determinar el número de alumnos a los cuales va dirigido el programa, lugar donde se lo va realizar, horario disponible, etc.

A continuación se asigna un coordinador del programa juntamente con un asesor, estas personas son las encargadas de elegir a los educadores y de presentar el presupuesto para el programa, luego se realiza una inscripción con las personas solicitantes del programa para luego matricularlas definitivamente cuando el programa este por empezar con el número definitivo de los alfabetizados que vayan asistir previo compromiso con las autoridades de la parroquia.

La asistencia se la controla en forma manual todos los días que haga clases, de la misma forma se controla a los desertores y también a los alfabetizados que ingresen una vez que el programa ya haya empezado hasta una fecha final determinada por el asesor de cada programa.

Todas estas novedades cada educador debe ingresar a hojas electrónicas cada fin de semana los días Viernes o Sábados, ya que luego el asesor deberá contabilizar estos datos y sacar los respectivos reportes. Todo este proceso demanda mucho tiempo en la actualidad y también genera inconvenientes puesto que si se necesita un reporte entre semana de la marcha del programa este no puede ser generado hasta la semana siguiente en que los asesores tienen acceso a la información y los educadores entregan sus informes.

Cada informe es almacenado en archivos de Excel sin las respectivas seguridades, ya que hace un tiempo se había dañado la maquina donde reposaban estos archivos perdiéndose valiosa información del programa sin ser recuperada en su totalidad, debiendo así ser ingresada nuevamente dicha información para generar los reportes de años anteriores.

### **1.3 Justificación.**

El presente estudio pretende profundizar en las tecnologías de código abierto que existen en el mercado para desarrollar aplicaciones Web, estas han sido muy eficientes desde todo punto de vista económico, puesto que existen herramientas que facilitan el desarrollo de las mismas, entre ellas utilizaremos las tecnologías PYTHON y PERL las mismas que han tenido una evolución muy lenta, en años anteriores, que en la actualidad han tenido notable evolución.

En este sentido el presente trabajo se considera relevante, por inscribirse en la línea de investigación del comportamiento de estas dos tecnologías que permitirá describir, comprender y aplicar en el desarrollo de una Aplicación Web.

La contribución que pretendemos los estudiantes de la Escuela de Ingeniería en Sistemas es el desarrollo de una Aplicación Web para el Programa de Alfabetización que lleva a cabo el Consejo Provincial de Chimborazo conjuntamente con el apoyo de la República de Cuba, el mismo que se realiza en nuestra provincia, puesto que en la actualidad existe un alto índice de analfabetismo con relación al resto de provincias.

Esta aplicación permitirá obtener los siguientes beneficios a la institución:

- Obtener información actualizada de las personas encargadas y del material didáctico que disponen los cantones, parroquias y comunidades, donde se ejecuta el Programa.
- Permitirá llevar un control (ingreso, actualización, eliminación, búsqueda) de los estudiantes matriculados en el Programa “Yo si Puedo”, de la Provincia.
- Obtención de reportes de los estudiantes matriculados y graduados, del Programa de Alfabetización “Yo Si Puedo” de la Provincia de Chimborazo.
- Además permitirá llevar un control cuantitativo del Programa en cada uno de los cantones.

## **1.4 Objetivos.**

### **1.4.1 Objetivo General**

- Realizar un Estudio Comparativo entre las Tecnologías de código abierto PYTHON y PERL para el desarrollo de Aplicaciones Web Implementado al Programa de Alfabetización del Consejo Provincial de Chimborazo (Periodo 2007 - 2008).

### **1.4.2 Objetivos Específicos**

- Estudiar las semejanzas, prestaciones y adaptaciones que facilitan las tecnologías de código abierto PYTHON y PERL para el desarrollo de Aplicaciones Web.
- Determinar criterios de selección que se adapten a nuestra Aplicación Web mediante pequeños módulos técnicos, los mismos que permitirán evaluar y analizar los beneficios de la utilización de las diferentes tecnologías (PYTHON Y PERL).
- Realizar una Aplicación Web que cumpla con los requerimientos establecidos por el departamento de Gestión social del Consejo Provincial de Chimborazo para el Programa de Alfabetización de nuestra Provincia.

### **1.5 Hipótesis**

Mediante el desarrollo de la Aplicación Web, se logrará evaluar y mejorar la Administración del Programa de Alfabetización que lleva a cabo el Departamento de Gestión Social del Consejo Provincial de Chimborazo, permitiendo el ahorro de recursos y ofreciendo a los usuarios un servicio eficiente.

## **CAPITULO II**

### **ASPECTOS GENERALES DE LAS TECNOLOGIAS PYTHON Y PERL**

#### **2.1 Lenguaje PYTHON**

##### **2.1.1 Introducción a PYTHON**

PYTHON es un lenguaje de programación interpretado e interactivo, capaz de ejecutarse en una gran cantidad de plataformas. Se desarrolla como un proyecto de código abierto, administrado por la PYTHON Software Foundation.

Fue creado por Guido van Rossum en 1990, creó durante unas vacaciones de navidad en las que al parecer se estaba aburriendo. Además creó como un lenguaje de programación de scripting, la "oposición leal" a PERL, lenguaje con el cual mantiene una rivalidad amistosa. Los usuarios de PYTHON consideran a éste mucho más limpio y elegante para programar. El nombre del lenguaje proviene de la afición de su creador original, Guido van Rossum, por los geniales humoristas británicos Monty PYTHON.

El principal objetivo que persigue este lenguaje es la facilidad, tanto de lectura, como de diseño.

Permite dividir el programa en módulos reutilizables desde otros programas PYTHON, viene con una gran colección de módulos estándar que se pueden utilizar como base de los programas. También hay módulos incluidos que proporcionan E/S de ficheros, llamadas al sistema, sockets y hasta interfaces a GUI (interfaz gráfica con el usuario).

Es un lenguaje interpretado, lo que ahorra un tiempo considerable en el desarrollo del programa, pues no es necesario compilar ni enlazar. El intérprete se puede utilizar de modo interactivo, lo que facilita experimentar con características del lenguaje, escribir programas desechables o probar funciones durante el desarrollo del programa, también es una calculadora muy útil.

PYTHON permite escribir programas muy compactos y legibles; los programas escritos en PYTHON son normalmente mucho más cortos que sus equivalentes en C o C++, por varios motivos:

- Los tipos de datos de alto nivel permiten expresar operaciones complejas en una sola sentencia.
- El agrupamiento de sentencias se realiza mediante sangrado en lugar de begin/end o llaves.
- No es necesario declarar los argumentos ni las variables.
- PYTHON es ampliable: si ya sabes programar en C, es fácil añadir una nueva función o módulo al intérprete, para realizar operaciones críticas a la máxima velocidad o para enlazar programas en PYTHON con bibliotecas que sólo están

disponibles en forma binaria (como bibliotecas de gráficos específicas del fabricante). Una vez enganchado, puedes enlazar el intérprete de PYTHON a una aplicación escrita en C y utilizarlo como lenguaje de macros para dicha aplicación.

En Mayo de 2000, Guido y el equipo principal de desarrollo de PYTHON se mudó a BeOpen.com para formar el equipo BeOpen PythonLabs. En Octubre del mismo año, el equipo PYTHONLabs se mudó a Digital Creations. En 2001, se formó la PYTHON Software Foundation, una organización sin ánimo de lucro, con el propósito específico de ser la propietaria de la propiedad intelectual relativa a PYTHON. Zope Corporation es patrocinador de la PSF.

Todas las versiones de PYTHON son Open Source. Históricamente, la mayoría, aunque no todas, de las versiones de PYTHON también han sido compatibles GPL; la siguiente tabla resume las diversas versiones:

**Tabla N° II. 1 Versiones de Python**

<b>Versión</b>	<b>Derivada</b>	<b>Año</b>	<b>Propietario</b>	<b>¿Compatible GPL?</b>
0.9.0 a 1.2	n/a	1991-1995	CWI	Sí
1.3 a 1.5.2	1.2	1995-1999	CNRI	Sí
1.6	1.5.2	2000	CNRI	No
2.0	1.6	2000	BeOpen.com	No
1.6.1	1.6	2001	CNRI	No
2.1	2.0+1.6.1	2001	PSF	No
2.0.1	2.0+1.6.1	2001	PSF	Sí
2.1.1	2.1+2.0.1	2001	PSF	Sí
2.2	2.1.1	2001	PSF	Sí
2.1.2	2.1.1	2002	PSF	Sí



2.1.3	2.1.2	2002	PSF	Sí
2.2.1	2.2	2002	PSF	Sí
2.2.2	2.2.1	2002	PSF	Sí
2.2.3	2.2.2	2002-2003	PSF	Sí
2.3	2.2.2	2002-2003	PSF	Sí
2.3.1	2.3	2002-2003	PSF	Sí
2.3.2	2.3.1	2003	PSF	Sí
2.3.3	2.3.2	2003	PSF	Sí
2.3.4	2.3.3	2004	PSF	Sí

**Nota:** compatible GPL no significa que se distribuya PYTHON bajo la GPL. Todas las licencias de PYTHON, a diferencia de la GPL, permiten distribuir una versión modificada sin hacer los cambios open source. Las licencias compatibles GPL permiten combinar PYTHON con otro software liberado bajo la GPL; las otras no.

## 2.1.2 Características de PYTHON

- **Muy legible y elegante.-** Imposible escribir código ofuscado.
- **Simple y poderoso.**
  - Muy denso: poco código hace mucho.
  - Soporta objetos y estructuras de datos de alto nivel: strings, listas, diccionarios, etc.
  - Múltiples niveles de organizar código: funciones, clases, módulos, y paquetes.
  - PYTHON Incluye library contiene un sinfín de clases de utilidad.

- Si hay áreas que son lentas se pueden reemplazar por plugins en C o C++, siguiendo la API para extender o empotrar PYTHON en una aplicación, o a través de herramientas.
- **De scripting**
  - No tienes que declarar constantes y variables antes de utilizarlas.
  - No requiere paso de compilación.
  - Alta velocidad de desarrollo y buen rendimiento.
- **Código interoperable**
  - Se puede utilizar en múltiples plataformas (más aún que Java)
  - Puedes incluso ejecutar Python dentro de una JVM (Jython)
- **Open source**
  - Razón por la cual la PYTHON Library sigue creciendo y creciendo
- **De propósito general**
  - Puedes hacer en PYTHON todo lo que puedes hacer con C# o Java.

### 2.1.3 Para que no es útil y si es útil PYTHON

- **PYTHON no es el lenguaje perfecto, no es bueno para:**
  - Programación de bajo nivel (system-programming), como programación de drivers y kernels.
  - PYTHON es de demasiado alto nivel, no hay control directo sobre memoria y otras tareas de bajo nivel.
  - Aplicaciones que requieren alta capacidad de cómputo.
  - No hay nada mejor para este tipo de aplicaciones que el viejo C
- **PYTHON es ideal para:**

- Como lenguaje “pegamento” para combinar varios componentes juntos.
- Para llevar a cabo prototipos de sistema.
- Para la elaboración de aplicaciones cliente.
- Para desarrollo Web y de sistemas distribuidos.
- Para el desarrollo de tareas científicas, en los que hay que simular y prototipar rápidamente.

#### **2.1.4 Programación Web en PYTHON**

Al tratarse PYTHON de un lenguaje de código abierto existen multitud de módulos/librerías para realizar cualquier tarea programática que imaginemos. En ocasiones, como es el caso de la programación Web, existen varios módulos diferentes que de una manera más básica o sofisticada nos permiten llevar a cabo la misma tarea. Para comprobarlo no tenemos más que visitar el portal de PYTHON, en su sección de Temas/Web <http://www.python.org/topics/web>. Allí podemos encontrar un largo listado de estos módulos, detallando sus diferentes funciones. A continuación enumeramos los más destacados, clasificados según la categoría de plataforma de desarrollo de aplicación Web a la que pertenecen.

El Módulo CGI de la librería Standard de PYTHON. CGI (Common Gateway Inteface) es un mecanismo estándar para la ejecución de código ejecutable por un servidor Web y la obtención de los resultados de tal ejecución. Módulo Cookie para la creación y procesamiento de cookies en aplicaciones Web. Una cookie es un mecanismo para mantener estado entre las peticiones HTTP de una sesión Web. Una cookie es una

cabecera HTTP que permite la identificación unívoca en el servidor Web del peticionario de la información.

En CGI, un nuevo proceso es creado por cada petición HTTP recibida y eliminado cuando la petición es resuelta. La eficiencia es pobre. Esta es la razón por la que han parecido numerosas tecnologías que permiten una integración superior con el servidor Web subyacente, y lo más importante, una mayor eficiencia. Ejemplos claros de estas tecnologías son PHP, Java Servlets y JSPs, y ASPs. En el caso particular de PYTHON, la contribución más interesante a este respecto es mod\_python: mod\_python es un módulo Apache que integra el intérprete PYTHON dentro del servidor, de modo que las aplicaciones pueden ejecutarse de manera más rápida que CGI, retiene datos persistentes entre las peticiones HTTP de una sesión y permite acceder a la parte interna de Apache.

#### **2.1.4.1 Servidores de aplicaciones.**

Van más allá de la simple generación de páginas dinámicas y asisten al programador en otras tareas de la programación de la lógica de negocio, tales como la persistencia de datos, la gestión de transacciones o la seguridad: Webware facilita una suite de componentes PYTHON para el desarrollo de aplicaciones Web.

Provee un servidor de aplicaciones, servlets, PYTHON Server Pages (PSP), mapeo de objetos a bases de datos relacional y organizador de tareas. La arquitectura es muy modular, y permite añadir tus propias extensiones. Zope, el Z Object Publishing Environment, proporciona una ORB (Object Request Broker) HTTP que permite

publicar e invocar objetos PYTHON en la Web sin necesidad de ningún tipo de código CGI o http específico. Objetos complicados pueden ser publicados con URLs que simulan jerarquías de objetos. Zope proporciona varios componentes que pueden usarse en concierto o separadamente. Algunos componentes incluidos son: un paquete de plantillas HTML, un sistema de persistencia de objetos, una plataforma para la gestión de objetos vía Web, e incluso un servidor Web sencillo.

#### **2.1.4.2 Integración Apache/PYTHON: mod\_python**

Grisha Trubetskoy (<http://www.ispol.com/home/grisha/>) creó mod\_python en 2003 con el propósito de ofrecer un módulo de extensión para el servidor Web Apache ([www.apache.org](http://www.apache.org)) que permitiese la generación de páginas dinámicas con PYTHON de una manera más eficiente que el tradicional módulo CGI. Mod\_python empotra el intérprete de PYTHON dentro del servidor Apache. Con mod\_python puedes escribir aplicaciones Web en PYTHON que ejecutan muchas veces más rápido que los CGIs tradicionales, tienen acceso a características avanzadas tales como retener conexiones a bases de datos entre conexiones HTTP y permiten acceder a la parte interna de Apache.

Con mod\_python, el código PYTHON se ejecuta directamente dentro del servidor Apache, sin necesidad de que éste tenga que lanzar procesos externos o delegar las peticiones a servidores de aplicaciones externos. Mod\_python se beneficia de las habilidades de Apache para aceptar y procesar peticiones entrantes de una manera escalable a la carga. El resultado es una plataforma que, según Trubetskoy, puede procesar peticiones más rápido que cualquier otra plataforma de desarrollo Web en PYTHON mod\_python es: Un módulo Apache recargable que empotra el intérprete de

PYTHON (libpython) proveyendo la habilidad de ejecutar código PYTHON en el mismo proceso que Apache. Un manejador de las fases de procesamiento de las peticiones HTTP en Apache, que permite implementar cualquier fase en PYTHON.

También permite implementar filtros y manejadores de conexiones en PYTHON. Un filtro HTTP intercepta de manera dinámica peticiones y respuestas para transformar o usar la información contenida en las mismas. Una interfaz a un subconjunto de las APIs de Apache, permitiendo la invocación de funciones internas del servidor desde PYTHON. Esto permite acceder a información interna del servidor o de beneficiarse de facilidades del mismo como logeo. Una colección de herramientas para el desarrollo de aplicaciones Web, provee un conjunto de manejadores de peticiones: Publisher (mapea URLs a objetos y funciones dentro de módulos PYTHON), PSP (permite la creación de Python Server Pages) y CGI (permite la programación Web conformando con el estándar CGI).

Cada uno de estos manejadores ofrece una manera diferente de desarrollar aplicaciones Web, así como un conjunto de objetos y funciones de utilidad para el procesamiento de cookies, gestión de sesiones, y otros aspectos comunes en el desarrollo Web. El mayor inconveniente de mod\_python es ser específico a Apache, a diferencia de JSPs y PHP que pueden integrarse con diferentes servidores Web cubiertos en la primera entrega de esta serie. Apache ha sido el servidor más popular en Internet desde 1996. En Octubre del 2003, 64% de los servidores Web en Internet usaban Apache. La mayoría de las distribuciones Linux ya traen preinstalado este servidor bien en su versión 1.3 o 2.0.

#### **2.1.4.2 Configuración Apache y Mod\_Python**

Una vez instalado Apache y mod\_python tan sólo resta modificar el principal fichero de configuración de Apache, ubicado en <directorioinstalación- apache>/conf/httpd.conf:

Al final del bloque de sentencias LoadModule añadir la siguiente: LoadModule python\_module modules/ mod\_python.so.

Después del bloque XML iniciado por el elemento Directory y correspondiente al directorio htdocs, directorio raíz del que cuelgan, por defecto, los documentos HTML estáticos en Apache, colocar los bloques Directory del listado 1, reemplazando <directorio- instalación-apache> por la ruta de instalación de Apache en tu máquina.

#### **2.1.4.3 Instalando Apache y mod\_python**

Para poder instalar mod\_python es requisito imprescindible la previa instalación del intérprete de PYTHON y el servidor Web Apache. Detalles sobre cómo instalar PYTHON tanto en Windows como UNIX fueron de instalación de Apache en tu máquina.

Con las definiciones mostradas anteriores indicaremos que debajo del subdirectorio python de cgi-bin, directorio por defecto del que cuelgan los scripts CGI en Apache, colocaremos programas PYTHON que generarán páginas HTML siguiendo el modelo del manejador de peticiones Publisher.

Con la segunda directiva Directory indicaremos que bajo el directorio psp colocaremos scripts en formato Python Server Pages (PSP).

#### **2.1.4.4 Python Server Pages en mod\_python**

PYTHON Server Pages (PSP) es un mecanismo para incluir sentencias PYTHON en documentos HTML o XML. El servidor interpreta el código PYTHON incluido para producir el documento HTML o XML enviado al cliente. Este mecanismo se ha hecho popular a través de otras herramientas bien conocidas como JSP, PHP o ASP. Es importante remarcar que mezclar código fuente PYTHON en mitad de un documento HTML es objeto de cierta controversia. Algunos ingenieros del software consideran incluir código en mitad de un documento HTML una mala práctica de programación, al violar el paradigma Modelo-Vista- Controlador, introduciendo lógica de negocio en la capa de presentación. A pesar de no ser tan buena práctica, los millones de programadores PHP que exitosamente utilizan este paradigma demuestran que existe gran demanda por este estilo de programación Web. La sintaxis de PSP es similar a la original de JSP, delimitando el código PYTHON por medio de los símbolos `<% y %>`.

De igual modo a JSP, PSP tiene cuatro tipos de entidades:

- **Código:** Representa el código fuente PYTHON que contiene la lógica de cómo la salida final es producida. Está delimitado por los códigos de escape `<% y %>`.
- **Expresiones:** Es código fuente PYTHON cuyo resultado en forma de un string forma parte de la salida final. Está delimitado por los códigos de escape `<%= y %>`.
- **Directivas:** Son instrucciones especiales para el procesador PSP. Son delimitadas por los códigos `<%@ y %>`. Actualmente `mod_python` sólo soporta la directiva `<%@ Incluye file='nombre-fichero'>`, que reemplazará esta directiva por el contenido del fichero 'nombre-fichero'.



- **Comentarios:** En PSP, son eliminados por el procesador PSP, y son delimitados por los símbolos `<%--` y `--%>`.

La parte más complicada de la programación PSP es la gestión de la tabulación sintáctica de PYTHON. El intérprete de PSP recuerda la última tabulación PYTHON usada a lo largo del código HTML, y deberemos ajustar nuestro código PYTHON a la misma.

Una página PSP (Python Server Page) es un documento HTML con instrucciones PYTHON embebidas en él que se interpretan como una plantilla para generar contenido dinámico. PSP es análogo a PHP, ASP o JSP. Para ello hace falta un servidor Web (como Apache) con el módulo `mod_python` instalado, configurar una serie de directivas en el archivo de configuración (de Apache) y ya podremos comenzar a mezclar código HTML con PYTHON.

Como podemos ver tenemos varias alternativas a la hora de desarrollar un sitio Web: con Zope podemos generar en poco tiempo portales orientados a la gestión de contenidos e información, con Webware podemos crear aplicaciones Web complejas al estilo de las que se pueden hacer utilizando j2ee y mediante las PSP podemos crear aplicaciones Web más sencillas también en poco tiempo.

#### **2.1.4.5 Sintaxis**

PSP es similar a JSP, delimitando su código usando la siguiente sintaxis: `<%` and `%>`

PSP tiene cuatro tipos de entidades:

Code: representa el código fuente PYTHON que conduce la lógica de cómo la salida final es producida. Es encerrado en los marcadores `<% and %>`.

Una Expresión es código PYTHON cuya cadena (string) resultante se convierte en parte de la salida final. Se delimitan así: `<%= and %>`

Las Directivas son instrucciones especiales para el procesador PSP. Las directivas se delimitan así: `<%@ and %>`

A diferencia de los comentarios HTML, los de PSP son removidos por el parser psp y nunca se encuentran en la salida final. Se delimitan así `<%-- and --%>`.

Esta sintaxis es simple y no cumple con XML, pero es un comienzo. Es necesario recordar que el código PYTHON debe estar indentado correctamente, por lo que es recomendable documentar las líneas para que sea legible. (Ver Anexo C).

Para usar PSP se debe configurar Apache y `mod_python` para usara el manejador `mod_python.psp`. Aquí esta la parte relevante del apache config:

```
<Directory /some/path>
    AddHandler mod_python .psp
    PythonHandler mod_python.psp
    PythonDebug On
</Directory>
```

El código JSP se encuentra en (Ver Anexo2). El archivo debe guardarse con extensión `.psp`.

Ese código PYTHON es compilado en código objeto usando la función `compile()` function. PSP agarra ese objeto y lo reutiliza para peticiones subsecuentes, a menos que el código fuente cambie en disco.

#### 2.1.4.5 Variables

Las variables no tienen tipo, así que no hay que declararlas. Las variables se escriben de forma dinámica, es decir, aparecen cuando se les asigna algo y desaparecen al dejar de usarlas.

La asignación se realiza mediante el operador `=`. Para comprobar la igualdad se utiliza el operador `==`. Se puede asignar varias variables a la vez: `x,y,z = 1,2,3` primero, segundo `=` segundo, primero.

```
A = b = 123
```

```
x = 1
```

```
x = 'texto' # esto es posible porque las variables son dinámicamente tipadas
```

#### 2.1.4.6 Tipos de datos

Tabla N° II. 2 Tipos de Datos en Python

Tipo	Clase	Ejemplo
Str	String	'Python'
unicode	String	u'Python'
list	Secuencia	[4.0, 'string', True]
tuple	Secuencia	(4.0, 'string', True)

set	Conjunto	set([4.0, 'string', True])
dict	Mapping	{'key1': 1.0, 'key2': False}
int	Número entero	42
long	Número entero	42L o 456966786151987643L
float	Número Coma flotante	3.1415927

## 2.2 Lenguaje PERL

### 2.2.1 Introducción a PERL

El aumento de sitios Web ha transformado el papel de PERL de un lenguaje de Script oscuro y desconocido a la herramienta principal de programación CGI. PERL, principalmente, es un lenguaje para la manipulación de cadenas de datos y de archivos, y su diseño ignora la distribución en pantalla de botones, menús, listas de selección, y otros elementos propios de una interfaz, es decir, no es nada vistoso, aunque su especialización en el uso de modelos de búsqueda lo hace muy potente.

PERL (Practical Extraction y Report Language) es un lenguaje de programación que se creó originalmente para extraer informes de ficheros de texto y utilizar dicha información para preparar informes. Actualmente ha evolucionado de forma que es posible realizar labores de administración en cualquier sistema operativo. Debe gran parte de su popularidad a tratarse de un intérprete que se distribuye de forma gratuita. Un script genérico de PERL puede ejecutarse en cualquier plataforma en la que tengamos un intérprete disponible.

Con el crecimiento del WWW se vio que era necesario realizar programas CGI y PERL se convirtió en la elección natural para los que ya estaban familiarizados con este lenguaje.

PERL es algo así como lenguaje práctico de extracción y de informes. Es un lenguaje creado por Larry Wall (quien, por cierto, es uno de los *net-gods* más conocidos en la actualidad) con el objetivo principal de simplificar las tareas de administración de un sistema UNIX. En realidad hoy en día, en su última versión, se ha convertido en un lenguaje de propósito general, y una de las principales herramientas de cualquiera que desee gestionar datos Web en Internet.

Es un lenguaje interpretado que tiene varias utilidades, pero está principalmente orientado a la búsqueda, extracción y formateado de ficheros de tipo texto. También es muy usado para manejo y gestión de procesos (estado de procesos, conteo y extracción de parámetros característicos, etc.).

Es una combinación de las características de los lenguajes más usados por los Programadores de sistemas, como son los shell del sistema operativo UNIX, los utilidad (que incluye un lenguaje interpretado propio) awk para formateo y tratamiento de texto e incluso características de Pascal, aunque su potencia se basa en la similitud con las mejores características del lenguaje estructurado C. En general cualquier utilidad que se necesite realizar en sh, awk, o sed, se puede implementar de una manera más potente y sencilla mediante el lenguaje PERL.

### **2.2.2 Características de PERL**

Existen varias características acerca de PERL que son importantes de rescatar, las cuales se mencionan a continuación:

- Es fácil de usar, aunque es difícil de aprender. Cuando se ha programado en PERL por varias horas, PERL se va a ir haciendo cada vez más fácil de implementar. Este lenguaje se desarrolló pensando en que el lenguaje fuera práctico (fácil de usar, eficiente, y completo) en lugar de pequeño, elegante y mínimo, además de que está enfocado hacia un desarrollador que posee cierta cantidad de conocimientos sobre el lenguaje y no así hacia un estudiante que está aprendiendo.
- Existen muy pocas cosas que no se pueden hacer con este lenguaje, por lo que se le puede considerar como un lenguaje que no tiene fronteras. Con PERL se puede programar cualquier necesidad que se tenga, ya que existen librerías y módulos para casi cualquier cosa que se requiera,
- Es rápido de crear, ya que no posee funciones que, aunque sean bastante interesantes, hagan disminuir la velocidad de desarrollo de una aplicación del lenguaje.
- El lenguaje es feo, siendo esta una de las principales razones para su difícil aprendizaje, aunque esto se compensa con el poder de alcance del lenguaje. Es por esto que Larry Wall escogió el camello como el logotipo del lenguaje, ya que aunque es feo, siempre trabaja fuerte aún en condiciones complicadas.
- Se puede utilizar en varios entornos, como puede ser Windows 95, OS/2, Linux, entre muchos otros, sin realizar cambios de código, siendo únicamente necesario la introducción del intérprete PERL correspondiente a cada sistema operativo.

- Tiene características que soportan una variedad de paradigmas de programación, como la estructural, funcional y la orientada a objetos. Al mismo tiempo, PERL no obliga a seguir ningún paradigma en particular, ni obliga al programador a elegir alguna de ellas. No obstante, esta característica es solo accesible en la versión 5.0.
- Tiene incorporado un poderoso sistema de procesamiento de texto y una enorme colección de módulos disponibles.
- Ofrece una ayuda en línea desde la consola de comandos. Por ejemplo, para obtener ayuda sobre la función `print`, se debe de escribir en una ventana MSDOS: `perldoc -f print`
- Se ejecuta desde la línea de comandos de una ventana del sistema operativo.
- Es un lenguaje *case-sensitive*, lo cual hace referencia a la propiedad de un texto para presentarse en mayúsculas o en minúsculas.

### **2.2.3 Programación Web con PERL**

En este modelo de programación, un programa o script o cgi en PERL puede ser una combinación de código y objetos HTML y código en PERL, Un objeto y sus métodos derivados de la clase CGI.PPY, esta clase ya casi un estándar para tratamiento y construcción de programas CGI con PERL, contiene una serie de métodos apropiados para la construcción de CGI's.

### **2.2.4 CGI's de Apache**

La interfaz CGI (Common Gateway Interface - interfaz general de ordenador entremedio) es una posibilidad de poner a disposición en la Web programas o scripts que pueden ser llamados desde archivos HTML y que ellos mismos pueden generar código HTML y enviarlo a un navegador Web. En contraste a JavaScript, estos scripts o programas no son ejecutados en el navegador del usuario después del envío, sino antes de que el navegador reciba los datos enviados por el servidor. Por eso, los scripts CGI o programas CGI sólo pueden ser ejecutados cuando hay una comunicación basada en HTTP entre el navegador Web y el servidor Web. En esta comunicación, CGI esta plantado en el lado del servidor.

El procesamiento de datos también es ejecutado en el servidor. Los programas CGI pueden guardar datos en el servidor, por ejemplo cuantas veces una página Web fue llamada o lo que un usuario escribió en un libro de visitas. Un script CGI puede también consultar bases de datos instalados en el servidor. Con una llamada correspondiente un programa CGI puede leer datos guardados y generar código HTML de ellos. Este código HTML generado "dinámicamente" es transmitido al navegador Web de un usuario que llama los datos y allí puede visualizar datos individuales en forma de HTML, por ejemplo el estado actual de un contador de acceso a una página Web, los mensajes en un libro de visitas hasta el tiempo actual o los resultados de una búsqueda en una base de datos.

No obstante, la mayoría de los programas CGI hoy en día no son programas compilados, sino solamente scripts los cuales son ejecutados por un intérprete en el momento de llamada.



PERL es un lenguaje de programación que puede ser descrito como una mezcla entre los lenguajes de programación clásicos como C y lenguajes de script como el script de shell de Unix. PERL es extremadamente hábil y el intérprete de PERL tiene un grado muy alto de madurez. Por eso, PERL conquistó los corazones de los programadores en innumerables campos. Por esta razón, la programación CGI y PERL entre tanto son mencionados en un aliento y presentados como congénere. Sólo es importante saber que el enlace entre CGI y PERL no es nada natural o necesario. CGI sólo es una norma para una interfaz de programación que debería ser soportada por el software de servidores Web y PERL es un lenguaje de script usable de modo universal que, sin duda, es muy apropiado para la programación CGI.

PERL no establece ninguna filosofía de programación (de hecho, no se puede decir que sea orientado a objetos, modular o estructurado aun cuando soporta directamente todos estos paradigmas de programación), los objetivos que se tuvieron en cuenta al diseñar la sintaxis de PERL fueron la facilidad de aprendizaje y de uso y la claridad de código, las cuales, considero que son necesarias (aunque pueden escribirse programas en PERL complejos e inteligibles si así se desea).

Por si fuese poco, PERL no es ni un compilador ni un intérprete, ya que está en un punto intermedio. Cuando mandamos a ejecutar un programa en PERL, se compila el código fuente a un código intermedio en memoria, se optimiza (como si fuésemos a elaborar un programa ejecutable) pero es ejecutado por un motor, como si se tratase de un intérprete. El resultado final, es que utilizamos algo que se comporta como un

intérprete pero que tiene un rendimiento comparativo al de programas compilados. Sin embargo, ya existen compiladores de PERL con la versión 5.8.x.

En fin, PERL no nos obliga a nada, pero como es lógico hay ciertas reglas que se recomienda seguir para facilitar nuestro trabajo:

- **Claridad:** En la mecánica de programación actual, los programas deben de ser entendibles por la persona que nos suceda en tareas de mantenimiento, de lo contrario perjudicamos tanto a nuestros compañeros de trabajo como a nuestra propia libertad para progresar y mantenernos libres de preocupaciones.
- **Indentación:** Una costumbre ya clásica de la programación, en lo personal, y a lo largo de los distintos programas, es tabular el código dos espacios hacia adelante al abrir cada bloque, y terminar la tabulación al terminar el bloque, de modo que las llaves de apertura y cierre quedan a la vista y en la misma columna, solas en sus renglones (esto incrementa algo el número de líneas, pero facilita sobremanera la búsqueda y corrección de los diversos bloques de control).
- **Nombres de Variables.** Es conveniente dar la máxima claridad a los nombres de las variables sin hacerlos demasiado grandes, el nombre de los contadores y variables que guardan valores concernientes a un pequeño segmento de código por lo regular son de un par de letras (c1, c2, ... cx para los contadores, s1, s2, etc para cadenas de entrada etc.) mientras que las variables que afectan a diversos segmentos (a modo de regla, que tienen su definición en una pantalla distinta a donde se usan) tienen nombres explicativos que conviene que no

excedan los 12 caracteres. Además, los nombres de archivos se usan con mayúsculas (ARCHENT, ARCHSAL, etc) y las clases tienen su primera letra mayúscula.

- **Comentarios:** Para facilitar la comprensión de un programa no hay nada mejor que explicarlo, y los comentarios son el medio ideal para hacerlo. Existen por lo menos tres comentarios que muchos consideran que siempre deben incluirse en un programa: qué hace el programa, quién lo escribió y cuándo inició y terminó de escribirlo, sobre todo en el contexto de una organización. Estos tres simples comentarios pueden hacer la diferencia entre desestimar un programa como indescifrable o dedicarle algún tiempo para revisarlo. Además, se considera prudente comentar dentro del código la forma en que el programa deberá ejecutarse, parámetros, y su sintaxis, así como comentar las estructuras de control como un modo de explicar la funcionalidad al detalle y resaltar con comentarios las funciones que cumplen las variables.
- **Sencillez:** Es cómodo en ocasiones el comprimir una serie de instrucciones en una sola línea, queda al criterio decidir cuando se gana en claridad con un código más o menos extenso, pero no debe titubearse en comentar el código que sea complejo.

### 2.2.5 Tipos de Datos en PERL

En el lenguaje PERL tiene tres tipos de datos:

- Escalares
- Vector de escalares
- Vectores asociativos de escalares

El tipo de dato escalar es denotado mediante un nombre de variable precedido del símbolo \$, y puede contener un número, una cadena de caracteres, y una cadena de caracteres en la que alguno de los caracteres puede ser un carácter especial.

El escalar, es la clase de datos que engloba los tipos convencionales de datos, de modo que podemos decir:

```
$v1="Pikachu";
```

```
$v1=100;
```

```
$v1=89.12;
```

Sin que esto implique ningún cambio en la naturaleza de \$v1, en todo momento es un escalar.

Aún cuando la compatibilidad de datos enteros y reales es fácil de imaginar, la conversión implícita en el caso de las cadenas no lo es, sin embargo la regla es bastante simple. Por ejemplo:

```
$v1="123y321";
```

Esto crea un escalar \$v1, que contiene la cadena "123y312". Pero ¿qué pasa si le deseamos sumar 4? PERL, interpreta \$v1 como entero (o punto flotante) para esto, toma todos los caracteres del inicio que forman un número correcto y ese número es el que interpreta; de modo que:

```
print $v1+1;
```

imprimirá124, y del mismo modo, como ejemplo:

```
$v2="123.123.123"+1 da el valor 124.123 a la variable $v1 (punto flotante).
```

Otro punto importante de las variables en general es que, aunque en ningún momento se declaran como de un tipo o de otro, si se pueden "destruir", o revisar que hayan recibido algún valor en la ejecución del programa, esto se logra mediante las funciones `defined()` y `undef()`: `defined` nos indica si la variable que le pasamos como argumento ha sido definida (es decir, existe en el programa) o no. `undef` toma una variable y la "elimina" de modo que ya no tiene valor y `defined` la reporta como no utilizada.

Los escalares son los constituyentes de las demás estructuras de datos, por lo que al explicar los arreglos, hashes, referencias y archivos haremos muchas referencias a ellos. Por ejemplo, si queremos definir una variable de tipo escalar llamada `numero` y que contenga un el número 12 haríamos:

```
$numero = 12;
```

Para almacenar una cadena de caracteres podemos hacer dos cosas:

- Definir el contenido entre comillas simples, no interpretándose ningún carácter especial contenido entre ellas, un ejemplo podría ser:

```
$palabra = 'pikachu';
```

- Definir el contenido entre comillas dobles, interpretándose cualquier carácter especial contenido entre ellas a la hora de la impresión de la cadena, un ejemplo podría ser:

```
$palabra = " pikachu \n";
```

Por último, podemos definir varias variables al mismo tiempo utilizando los paréntesis por ejemplo:

```
($palabra1,$palabra2,$palabra3) = ("Pikachu","Espinete","Don Pimpón");
```

El tipo de dato vector de escalares es denotado mediante un nombre de variable precedido del símbolo @. Un ejemplo de inicialización de una variable de este tipo podría ser:

```
@frutas = ("manzana","pera","naranja","fresa");
```

El número de elementos menos uno de un vector de escalares se puede obtener mediante el nombre de la variable precedido de los símbolos \$# , por ejemplo si queremos obtener el número de elementos de @frutas haríamos:

```
$numero_de_elementos = $#frutas+1;
```

Para ser más exactos \$# nos devuelve el último índice del vector, así, si el valor para el primer índice es 0 el número de elementos - 1 será el último valor permitido para indexar el vector.

El acceso a un elemento de un vector se hace mediante un índice numérico de la forma \$nom\_variable [ indice ] ; así, si quisiéramos acceder al tercer elemento de

```
@frutas haríamos:
```

```
$2_elemento = $frutas[2];
```

También podemos acceder a varios elementos de un vector de la forma siguiente:

```
@nom_variable[indice1,...,indicen]
```

Veamos ahora otra instrucción:

```
print @frutas[1,2];
```

Esto imprimiría el elemento segundo y tercero del vector @frutas. Otra forma de indexar un vector es usando el operador rango, si quisiéramos imprimir todos los valores desde un índice i hasta un índice j deberíamos utilizar el siguiente formato

@nom\_variable[i..j], por ejemplo:

```
print @frutas[0..2];
```

Así se imprimirían todos los elementos del vector @frutas. Por último tenemos también el tipo de dato vector asociativo de escalares, mediante este tipo de datos podemos acceder a un determinado elemento de un vector a través de una clave anteriormente definida, para denotar este tipo de dato debemos preceder al nombre de la variable con el símbolo %. Un ejemplo de definición de claves y valores de un determinado vector asociativo podría ser:

```
%precio = ( "manzana",100,"pera",20,"naranja",30,"fresa",200 );
```

Si queremos acceder a un elemento de un vector asociativo, debemos seguir el siguiente formato @nom\_variable { clave }, un ejemplo podría ser:

```
$numero_de_manzanas=10;
```

```
$gasto = @precio {"manzana"} * $numero_de_manzanas;
```

### **2.2.6 Sintaxis en PERL**

Como ya hemos podido observar, todas las instrucciones de PERL deben acabar en ;(al igual que en C y C++) .Por ejemplo:

```
$numero = 2; print $numero;
```

Además, las únicas cosas que deben de ser declaradas en PERL son los formatos de salida o reports y las subrutinas. Todas las variables creadas y no inicializadas tienen como valor por defecto nulo o cero.

### 2.2.7 Sentencias de Control

Ahora veremos de forma práctica el uso de las sentencias de control condicionales e iterativas que suministra el lenguaje PERL para alterar el flujo de un programa.

- En primer lugar entenderemos como bloque a todo conjunto de sentencias que se encuentren entre dos llaves, por ejemplo lo siguiente sería un bloque:
  - `{`
  - `$numero=2;`
  - `print $numero;`
  - `}`
- Los bloques, además, pueden ser etiquetados anteponiendo a la llave de apertura el nombre de la etiqueta más dos puntos.
- En segundo lugar entenderemos como expresión a toda expresión cuya evaluación resulte ser un valor booleano, es decir, verdadero o falso. Hay que tener en cuenta que en PERL se considera como verdadero todo valor no nulo, luego se pueden presentar como expresiones el resultado de una asignación, de una búsqueda etc.

Una vez hechas estas consideraciones pasamos a ver la sintaxis de los distintos esquemas de control, que como ya hemos dicho antes pueden ser:



- Esquemas condicionales.
- Esquemas iterativos.

Dentro de los primeros tenemos los siguientes formatos:

**if (expresión) bloque.**

Un ejemplo se lo puede encontrar en el Anexo C:

En lugar de if podemos usar unless que tiene el significado contrario. Dentro de los esquemas iterativos tenemos:

**while (expresión) bloque.**

Mientras se cumpla la condición EXP se ejecutará todo lo que haya dentro del bloque. Un ejemplo se encuentra en Anexos:

En lugar de while podemos usar until, que tiene el efecto contrario a éste. Si además queremos hacer la primera comparación después de la primera iteración, podemos usar un esquema iterativo do BLOQUE until o do BLOQUE while.

**for (expresión; expresión; expresión)**

Su uso es exactamente igual que en C; un ejemplo de uso sería:

```
print "IMPRIMIENDO DEL 1 al 100:\n";  
  
getc;  
  
for($i=1;$i<101;$i++)  
{  
  print $i."\n";  
}
```

**foreach variable (array) bloque.**

Con este esquema vamos obteniendo cada uno de los valores que contiene ARRAY y los vamos depositando en la variable VAR. Un ejemplo podría ser:

```
@frutas=("manzana","pera","naranja","fresa");  
  
foreach $fruta (@frutas)  
{  
    print $fruta."\n";  
}
```

PERL soporta una gran cantidad de operadores, la mayoría de ellos heredados del lenguaje C. En PERL los operadores conservan el mismo uso que en el resto de lenguajes, y se utilizan para realizar operaciones aritméticas, operaciones lógicas, entre las variables del programa.

EL procedimiento de construcción de programa construye una forma o página dinámica que contiene los componentes, controles u objetos HTML para resolver el problema, y un poco de código en PERL para resolver el problema planteado.

## **2.3 PERL vs PYTHON**

### **2.3.1 Ventajas y Desventajas de PERL y PYTHON**

#### **2.3.1.1 Ventajas de PERL**

- Es un buen lenguaje ``pegamento". Se pueden juntar varios programas de una forma sencilla para alcanzar una meta determinada. Los usuarios de Windows

agradecen esta propiedad ya que normalmente adolecen de un buen lenguaje tipo ``script".

- Es relativamente rápido para un lenguaje tipo ``script"
- Esta disponible en múltiples plataformas y sistemas operativos. De hecho funciona bajo diferentes sabores de UNIX, Linux y todo tipo de Windows. Un programa que se escriba teniendo en cuenta la compatibilidad puede ser escrito en una plataforma y ejecutado en otra
- El desarrollo de aplicaciones es muy rápido.
- Hay una colección enorme de módulos que pueden ser incorporados a cualquier ``script" de Perl. Están disponibles en el CPAN (``Comprehensive Perl Archive Network"). En particular existe una extensión para cálculo numérico denominada PDL.
- Perl es gratuito. Mucho mas que eso, es ``Software Libre". Esto quiere decir que el código fuente disponible para que cualquiera lo pueda ver o modificar, y lo que es mas importante, siempre lo estará. Aunque nunca pretendas cambiar el código, es importante disponer de la posibilidad de hacerlo, ya que siempre se podrá contratar a una tercera persona para que lo modifique en el caso de que haya un error, y debería ser posible solucionarlo.
- Le otorga al programador mucha libertad para que haga el programa como quiera. Tal como dice el eslogan de Perl. Hay más de una forma de hacerlo.

### **2.3.1.2 Desventajas de PERL**

- Es lento para algunas aplicaciones, como programación a bajo nivel, escribiendo un "driver" para una aplicación o corriendo modelos numéricos de cálculo intensivo.
- La libertad que se le otorga al programador puede significar que el resultado sea un programa ilegible. Si no se escribe con cuidado puede llegar a ser difícil de leer.
- Utiliza muchos recursos de la máquina. Esto significa que no es tan ligero como un programa en C, pero en la práctica es ligero comparado con la potencia de computación de los ordenadores actuales.

#### **2.3.1.3 Ventajas de PYTHON**

- Se aprende excesivamente rápido
- Se puede hacer casi cualquier tipo de cosa con este lenguaje
- Tiene orientación a objetos integrada en el lenguaje
- Tiene la posibilidad de integrar con otros lenguajes
- Tiene una multitud de APIs y librerías para una gran cantidad de usos
- Posee un código limpio y fácil de entender
- Es un lenguaje de muy alto nivel
- Posee un gran manejo de estructura de datos
- Tiene funciones de programación funcional.

#### **2.3.1.4 Desventajas de PYTHON**

- No tiene aplicaciones programadas para un sistema operativo específico.

- PYTHON aun no tiene GUI independiente con sus versiones descargables, ni un IDE que compita con los IDEs de otros lenguajes.
- Además es casi imposible encontrar aplicaciones en PYTHON puro sin depender de otros lenguajes.
- No existe ningún tutor desarrollado en PYTHON.
- Tiene poca actualidad ya que depende siempre del avance en otros lenguajes, faltan proyectos de investigación en PYTHON, que permitan darlo a conocer o ofrecer nichos propios, aun falta explotar esta ventaja, como hacer aplicaciones de nuevas tecnologías para hardware viejo.
- PYTHON es universal, pero no se ven muchas aplicaciones sobre Windows, Mac, o dispositivos móviles. Actúa como lenguaje integrador, como se le ve en OpenOffice, o como lenguaje de script en muchas Web.
- PYTHON es poco difundido como lenguaje.

### **2.3.2 Semejanzas y diferencias entre PYTHON y PERL**

#### **2.3.2.1 Semejanzas entre PERL y PYTHON**

- Una lista en PYTHON es como un array en PERL. En PERL, las variables que almacenan en arrays comienzan siempre con el carácter @; en PYTHON, las variables pueden llamarse de cualquier modo, y PYTHON lleva el registro del tipo de datos internamente.
- Las triples comillas son también una forma fácil de definir una cadena que incluya comillas simples y dobles.
- Import en PYTHON es como require en PERL. Cuando se ha importado un módulo en PYTHON, se puede acceder a sus funciones con *módulo.función*;

cuando se ha requerido un módulo en PERL, se puede acceder a sus funciones con *módulo::función*.

- Un diccionario en PYTHON es como un hash en PERL. En PERL, las variables que almacenan hashes comienzan siempre por el carácter %; en PYTHON, las variables pueden tener cualquier nombre, y PYTHON guarda internamente el tipo de dato.
- Cada función de PYTHON puede realizarse fácilmente en PERL
- Los dos lenguajes tienen desarrollo de prototipos y son interpretados
- Ambos lenguajes son muy maduros y tienen código fuente publicado y abierto.
- Son lenguajes libres
- Tienen compatibilidad amplia, actualizaciones gratis, y manuales gratis en la Web.

### **2.3.2.2 Diferencias**

- PYTHON tiene chequeo dinámico de tipos y PERL no tiene chequeo de tipos
- PERL tiene facilidad de tratamientos de strings y expresiones regulares en cambio para PYTHON es complicado.
- PYTHON tiene más desarrollo de OO pero PERL no tiene muy desarrollado.
- PERL tiene una base más grande de módulos abiertos que PYTHON.

### **2.3.2.3 Fortalezas y Debilidades de Perl y Python**

Para la determinación de las fortalezas y debilidades de cada tecnología realizaremos un análisis tipo FODA, que consiste en un resumen de las ventajas, desventajas, oportunidades y amenazas de cada tecnología.

**Tabla N° II. 3 Análisis Perl**

<b>ANALISIS PERL</b>	<p style="text-align: center;"><b>FORTALEZAS</b></p> <p>Es un lenguaje pegamento</p> <p>Es relativamente rápido</p> <p>Esta disponible en múltiples plataformas</p> <p>Ofrece libertad al programador</p>	<p style="text-align: center;"><b>DEBILIDADES</b></p> <p>Lento para algunas aplicaciones como programación a bajo nivel</p> <p>La libertad que otorga al programador puede ocasionar programas ilegibles</p>
<p style="text-align: center;"><b>OPORTUNIDADES</b></p> <p>- Es un lenguaje Libre.</p> <p>- Muchos programadores pueden aportar para su mejoramiento.</p> <p>- Fácil de conseguir.</p> <p>- Abundante información</p>	<p style="text-align: center;"><b>ESTRATEGIAS FO</b></p> <p>Utilidad máxima porque utiliza módulos.</p> <p>En Linux sus módulos ya vienen instalados y listos para ser utilizados simplemente hay que levantarlos.</p>	<p style="text-align: center;"><b>ESTRATEGIAS DO</b></p> <p>Incentivar a los programadores a encontrar mejores mecanismos para la programación a bajo nivel.</p>
<p style="text-align: center;"><b>AMENAZAS</b></p> <p>- Lenguaje poco entendible.</p> <p>- Lenguaje maduro</p>	<p style="text-align: center;"><b>ESTRATEGIAS FA</b></p> <p>Privilegiado para realizar un trabajo en grupos</p>	<p style="text-align: center;"><b>ESTRATEGIAS DA</b></p> <p>Ha servido de partida para la creación de otros lenguajes</p> <p>Es un lenguaje critico de los demás.</p>

**Tabla N° II. 4 Análisis Python.**

<b>ANALISIS PYTHON</b>	<p style="text-align: center;"><b>FORTALEZAS</b></p> <p>Aprendizaje rápido</p> <p>Tiene orientación a objetos</p> <p>Integración con otros lenguajes</p> <p>Multiplataformas</p> <p>Tiene una multitud de APIs</p>	<p style="text-align: center;"><b>DEBILIDADES</b></p> <p>No tiene GUI independiente con sus versiones.</p> <p>Difícil encontrar aplicaciones con Python puro sin depender de otros lenguajes</p> <p>Depende del avance de otros lenguajes</p> <p>No actúa en bastantes aplicaciones</p> <p>Es poco difundido.</p>
<p style="text-align: center;"><b>OPORTUNIDADES</b></p> <p>- Es Open Source.</p> <p>- Sintaxis comprensible.</p> <p>- Programación alto nivel</p>	<p style="text-align: center;"><b>ESTRATEGIAS FO</b></p> <p>Facilidad de uso.</p> <p>En Linux sus módulos ya vienen instalados.</p>	<p style="text-align: center;"><b>ESTRATEGIAS DO</b></p> <p>Difundir más acerca de la potencialidad de este lenguaje.</p>
<p style="text-align: center;"><b>AMENAZAS</b></p> <p>- Presenta menores facilidades con relación a Ruby y Java .</p>	<p style="text-align: center;"><b>ESTRATEGIAS FA</b></p> <p>Promueve el mecanismo de la reutilización de componentes</p>	<p style="text-align: center;"><b>ESTRATEGIAS DA</b></p> <p>Esta en evolución con la aparición de Python.Net .</p>

## **CAPITULO III**

### **ANÁLISIS COMPARATIVO DE LAS TECNOLOGIAS PERL Y PYTHON PARA DESARROLLAR APLICACIONES WEB**

#### **3.1 Introducción**

La determinación de la tecnología Web que se va a utilizar para el desarrollo de un sistema informático, es una de las decisiones mas importantes que deben tomar los desarrolladores, esta decisión debe estar basada en un profundo y minucioso análisis de acuerdo a criterios de comparación o parámetros, además que debe cumplir tanto con los estándares que requieren las aplicaciones Web en la actualidad, como también con los requerimientos de la institución para la cual se va a desarrollar el sistema informático.

Los criterios utilizados para el análisis comparativo deben tener como base las características más importantes de las tecnologías, por lo que se tomaran en cuenta los aspectos que presenten mayor relevancia para tomar una decisión.

En este capitulo se realizará un análisis comparativo entre 2 tecnologías de código libre para seleccionar la tecnología mas adecuada para el desarrollo de aplicaciones Web,



acorde a las necesidades del usuario, los resultados obtenidos del presente estudio se expresan en forma de conclusiones y recomendaciones.

### **3.2 Determinación de las Tecnologías a Comparar**

En la actualidad existen un gran número de tecnologías para desarrollar aplicaciones Web, que son bastante populares, y de gran importancia para los desarrolladores, de las cuales se han seleccionado 2 tecnologías por las razones que se explican a continuación: Esta comparación deberá ser realizada con tecnologías de programación Web que sean conocidas, fáciles de conseguir y que en la actualidad gocen de mejoras notables, es por esto que para la realización de este análisis comparativo se han seleccionado a las tecnologías de desarrollo Web PERL y PYTHON.

Tanto PERL y PYTHON gozan de una gran popularidad entre los desarrolladores Web, además que cuentan con una amplia gama de recursos disponibles para su utilización.

Además las dos tecnologías tienen libre distribución, y brindan más y mejores opciones a los desarrolladores, ya que son lenguajes gratuitos y mucho más fáciles de conseguir.

### **3.3 Análisis de las Tecnologías Seleccionadas**

A continuación realizaremos un análisis de las tecnologías de desarrollo Web seleccionadas.

#### **3.3.1 PERL**

Perl es un lenguaje de propósito general, el cual fue originalmente desarrollado para extraer informes de ficheros de texto y utilizar dicha información para preparar informes.

No obstante, actualmente ha evolucionado y se ha diseccionado hacia un enfoque diferente con el que se creó el lenguaje, siendo capaz de realizar labores de administración en cualquier sistema operativo, tales como administración de sistemas, desarrollo Web, programación en red, desarrollo de GUI(*Graphical User Interface*), así como otras aplicaciones prácticas.

Este lenguaje debe gran parte de su popularidad a que se trata de un lenguaje pseudo-compilado que se distribuye de forma gratuita; un Script genérico de PERL puede ejecutarse en cualquier plataforma en la que tengamos un intérprete disponible.

Además, con el crecimiento acelerado de sitios Web, se generó la necesidad de realizar programas CGI (*Common Gateway Interface*) el cual define un protocolo de comunicación entre un servidor Web y una aplicación externa para ofrecer contenido dinámico a las páginas Web; con lo que PERL se convirtió en la elección natural para aquellos desarrolladores que se encontraban familiarizados con este lenguaje.

PERL es una combinación de las características de los lenguajes más usados por los programadores de sistemas, como son los *shell* del sistema operativo UNIX, los utilidad (que incluye un lenguaje interpretado propio) awk para formateo y tratamiento de texto e incluso características de Pascal, aunque su potencia se basa en la similitud con las mejores características del lenguaje estructurado C.

Para trabajar con PERL se requieren dos requerimientos básicos y fáciles de conseguir:

- 1.- Un editor de texto para poder escribir los programas PERL.
- 2.- El intérprete de PERL, que ejecute los programas realizados con PERL.

### **3.3.1.1 Manipulación de base de Datos con PERL**

Cuando utilizamos Perl podemos acceder a una base de datos por medio del modulo DBI.

DBI, es un módulo (librería o conjunto de funciones) de PERL que se especializa en procesar bases de datos de diversas compañías, una de las principales ventajas de este modulo DBI es su portabilidad, es decir si se construyen sistemas de base de datos en MySql, el mismo sistema PERL se puede usar para ORACLE, SYSBASE, etc. con muy pequeños cambios en el código original.

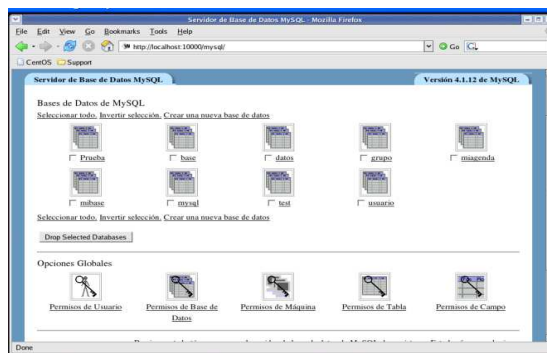
Lo más fácil es usar el módulo DBI, con un driver específico para el gestor de bases de datos, o si hay ninguno específico, y la base de datos está accesible usando ODBC, puede utilizarse con ODBC. En Windows hay un módulo alternativo, específico para ODBC, denominado Win32::ODBC.

Estos dos módulos, se puede usar con cualquier sistema de gestión de bases de datos. En Linux y UNIX, se puede usar con MySQL o MiniSQL, o Postgres. Por último, en Windows se puede usar también Microsoft SQL Server combinado con ODBC. Eso sí, como es inherente a las conexiones ODBC, es una forma menos ágil de gestionar una base de datos.

Como se mencionó anteriormente, PERL no obliga a casi nada, así pues, lo que se plantea como estructura básica es más bien una convención que un requisito del lenguaje, a diferencia de Pascal (por ejemplo) PERL no tiene una plantilla para sus programas. Si se adoptan algunos protocolos es sólo por comodidad.



Una vez situados en el servidor de datos de MySQL procedemos a realizar una Base con el Nombre “consejo” con una tabla llamada “cantones” y así ya podemos utilizar la base de datos en la Web.



**Figura N° III. 2 Tablas de la Base de Datos**

Las operaciones fundamentales como inserción, eliminación, actualización y consulta de datos en una base de datos, en PERL se realiza con sintaxis , aunque un poco compleja de entender, pero eficiente a la hora de obtener resultados

Dado a su amplia gama de módulos funcionales, es posible su portabilidad entre sistemas operativos, sin mayores inconvenientes para el usuario, ya que es un lenguaje OpenSource. En este aspecto Perl presenta facilidades ya que podemos levantar aplicaciones en sistemas operativos como Linux, Unix, Windows etc, sin mayores inconvenientes. Así mismo el módulo de base da datos que utiliza permite interactuar con diferentes motores de base de datos.

En los últimos tiempos PERL ha evolucionado notablemente, principalmente presentando, a los programadores interfaces de desarrollo mas amigables y con mas

funcionalidades, para el desarrollo de aplicaciones. Facilitando de esta manera el trabajo para los programadores.

### **3.3.2 PYTHON**

PYTHON es un lenguaje de programación interpretado e interactivo, capaz de ejecutarse en una gran cantidad de plataformas, el principal objetivo que persigue este lenguaje es la facilidad, tanto de lectura, como de diseño; permite dividir el programa en módulos reutilizables desde otros programas PYTHON.

Viene con una gran colección de módulos estándar que se pueden utilizar como base de los programas. También hay módulos incluidos que proporcionan E/S de ficheros, llamadas al sistema, sockets y hasta interfaces a GUI (interfaz gráfica con el usuario).

Es un lenguaje interpretado, lo que ahorra un tiempo considerable en el desarrollo del programas y aplicaciones. PYTHON permite escribir programas muy compactos y legibles. Los programas escritos en PYTHON son normalmente mucho más cortos que sus equivalentes en C o C++.

PYTHON es ampliable: si ya sabes programar en C, es fácil añadir una nueva función o módulo al intérprete, para realizar operaciones críticas a la máxima velocidad o para enlazar programas en PYTHON con bibliotecas que sólo están disponibles en forma binaria (como bibliotecas de gráficos específicas del fabricante). Una vez enganchado, puedes enlazar el intérprete de PYTHON a una aplicación escrita en C y utilizarlo como lenguaje de macros para dicha aplicación.

#### **3.3.2.1 Manipulación de Base de Datos con PYTHON.**

Al igual que Java que ofrece el estándar JDBC para el acceso a bases de datos, PYTHON hace lo propio con su DB-API. Con esta API PYTHON consigue aislar el código fuente de la base de datos subyacente. El código PYTHON se acomodará a los interfaces de la DB-API, mientras que una implementación de esta API para cada sistema de bases de datos, traducirá invocaciones de DB-API en llamadas de la base de datos.

Para conectarnos a una base de datos usamos el método connect del módulo de base de datos utilizado que devuelve un objeto de tipo connection.

El objeto connection ofrece el método cursor() que sirve para recuperar un cursor de la BD. Otros métodos definidos en connection son close(), commit(), rollback() y cursor(). El objeto cursor define entre otros los siguientes métodos: execute() nos permite enviar una sentencia SQL a la BD. fetchone() recupera una fila. \_fetchall() recuperar todas las filas.

Como se ha comentado, cada sistema relacional de bases de datos ha de ofrecer una implementación de DB-API. Dada la naturaleza de código abierto de Python, hay varios módulos que implementan este estándar:

Por su parte, la instalación de MySQLdb, implementación para MySQL de la DB-API, es también trivial. Simplemente accede a su página web (<http://sourceforge.net/projects/mysql-python>) y usa el instalador provisto para Windows o el RPPY para Linux.

### **3.3.3 Instalación y Configuración de Perl y Python en Linux Centos 4.3**

Para poder construir programas o aplicaciones orientadas a Internet en PERL CGI Y PYTHON para Linux, se ocupa lo siguiente:

- Un computador (pc) que se usa como servidor, el sistema PYTHON Y PERL con sus respectivos módulos de ejecución que ya contiene Linux, como nota importante es entender que Linux tiene muchas distribuciones diferentes y cada una de ellas esta configurada de varias maneras, la que se esta usando en esta aplicación es el LINUX CENTOS 4.3.
  
- Un programa o aplicación llamado servidor de paginas (Web Server) que debe estarse ejecutando todo el tiempo en el servidor físico.  
Linux Centos 4.3 (sistema PERL operativo) ya trae integrado el servidor de páginas APACHE ADVANCED EXTRANET SERVER.
  
- El mejor editor del mundo para crear paginas HTML y programas de PERL, que hemos utilizado en nuestra aplicación es el PERL EXPRES 2.5, ACTIVE PYTHON 4.1 NOTEPAD Y DREAMWEAVER MX 2008 de Windows, con el cual editamos la scripts tanto de PERL y PYTHON con esto los programas se crearan en una maquina Windows y se subirán al servidor Linux-apache.

PYTHON para la inserción, eliminación, actualización y consulta de datos en una base de datos, utiliza una sintaxis elegante, que permite ser entendida y mejorada por los programadores.



Dado que es un lenguaje Open Source y debido a que utiliza módulos funcionales su portabilidad entre sistemas operativos es posible, sin mayores inconvenientes para el usuario. Aspecto que presenta facilidades ya que podemos trabajar y levantar aplicaciones en sistemas operativos como Linux, Unix, Windows etc,

En los últimos tiempos PYTHON ha tenido una evolución notable, presentando a los programadores interfaces de desarrollo amigables y con mas funcionalidades, para el desarrollo de aplicaciones. Facilitando de esta manera el trabajo para los programadores.

### 3.3.3.1 Configuración de los Módulos e PERL y PYTHON en Apache.

Los módulos tanto de PYTHON y PERL se encuentran ubicados en el directorio CONF o carpeta, que se encuentra dentro de /etc/httpd/ en el cual se encuentra un archivos plano HTTPD.COMF de configuración del Apache el cual modificamos algunas líneas de control.

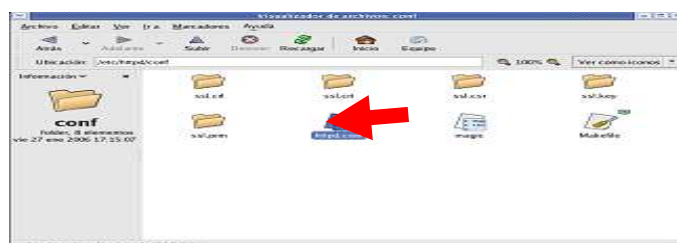


Figura N° III. 3 Configuración de Módulos Perl y Python en Linux

Una vez localizado aquel archivo plano de HTTPD.COMF del Apache procedemos a abrir en un editor de texto de Linux el Gedit, escribir las siguientes sentencias para la

ejecución o se carguen los módulos de PYTHON y PERL (mod\_python, mod\_perl). En la cabecera del documento.



```
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule cache_module modules/mod_cache.so
LoadModule suexec_module modules/mod_suexec.so
LoadModule disk_cache_module modules/mod_disk_cache.so
LoadModule file_cache_module modules/mod_file_cache.so
LoadModule mem_cache_module modules/mod_mem_cache.so
LoadModule cgi_module modules/mod_cgi.so

#configuracion de modulos de python y perl en el apache
LoadModule python_module modules/mod_python.so
LoadModule perl_module modules/mod_perl.so
```

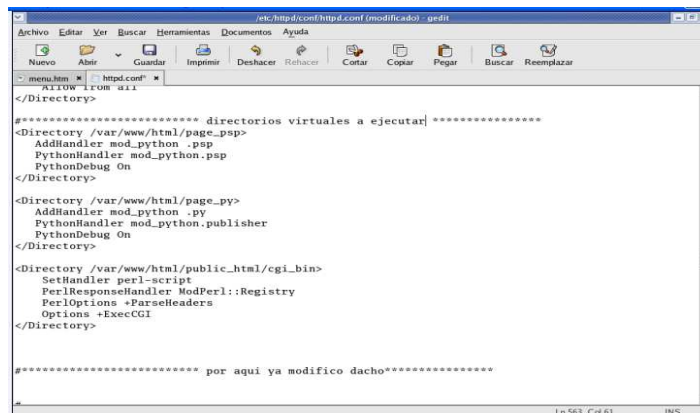
Figura N° III. 4 Módulos de Python en Linux

Otro punto muy importante es la asignación de los **sitios virtuales en el Apache** donde se van a ejecutar nuestras páginas en la Web lo cual hacemos tres directorios virtuales los cuales son:

**page\_psp** .Donde se ejecutan los scripts de PYTHON .PSP

**page\_py**. Donde se ejecutan los scripts de PYTHON.PY

**cgi-bin**. Donde se ejecutan los scripts de PERL .Pl



```
</Directory>
#***** directorios virtuales a ejecutar*****
<Directory /var/www/html/page_psp>
  AddHandler mod_python .psp
  PythonHandler mod_python.psp
  PythonDebug On
</Directory>

<Directory /var/www/html/page_py>
  AddHandler mod_python .py
  PythonHandler mod_python.publisher
  PythonDebug On
</Directory>

<Directory /var/www/html/public_html/cgi_bin>
  SetHandler perl-script
  PerlResponseHandler ModPerl::Registry
  PerlOptions +ParseHeaders
  Options +ExecCGI
</Directory>

#***** por aqui ya modifico dacho*****
```

Figura N° III. 5 Asignación de Sitios Virtuales

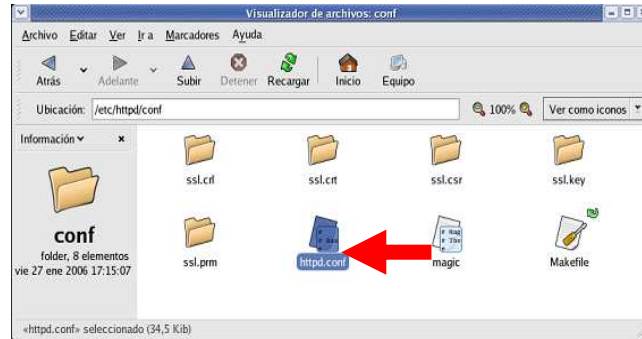
A continuación creamos tres carpetas dentro del siguiente directorio **/var/www/html/** donde se publican por defecto las páginas html en el Apache los cuales son:

**page\_psp** .Donde se ejecutan los scripts de PYTHON .psp

**page\_py**. Donde se ejecutan los scripts de PYTHON.py

**html\_public** y dentro de este otro el **cgi-bin** Donde se ejecutan los scripts de PERL .pl.

Los módulos de PYTHON y PERL son mod\_python y mod\_perl a ejecutarse en el Apache se encuentran ubicados en el siguiente directorio dado en el siguiente grafico.



**Figura N° III. 6 Ubicación de los Módulos Perl y Python**

Por último en la configuración del Apache ejecutamos en una consola el siguiente comando **service httpd restart** y así restablecemos el servicio httpd para que cargue la nueva configuración realizada anteriormente, lo cual nos comunica que los módulos ya se cargaron satisfactoriamente.



**Figura N° III. 7 Verificación de Módulos Cargados**

El mod\_python y mod\_perl permite cargar automáticamente en el Apache los siguientes módulos que se encuentra por defecto en el Linux Centos los mas importantes son MySQLdb,HTML parser para PYTHON y DBI, CGI para PERL que controla la base de datos MySql y la ejecución en la Web.



Figura Nº III. 8 Módulos Perl y Python

### 3.4 Determinación de los Parámetros de Comparación

Los parámetros, que a continuación, se definirán para la realización del estudio comparativo de las Tecnologías Perl y Python están basadas en un artículo (Recopilación de los requerimientos básicos de un sitio Web) publicado por James Robertson en el cual indica algunos de los requerimientos básicos que debe tener una aplicación Web, y también por los autores de esta tesis.

Los criterios que se va a considerar en general en este estudio, para analizar las dos tecnologías para el desarrollo de aplicaciones Web son:

#### 3.4.1 Acceso a la Base de Datos

Este parámetro es muy importante dentro de las aplicaciones Web porque permite acceder y manipular información almacenada en una bases de datos, a través de la utilización de consultas, las mismas que deben ejecutarse de una forma que sea lo más rápida y segura posible.

#### 3.4.2 Líneas de Código

Este parámetro nos permitirá de forma práctica obtener el lenguaje que ocupa menor cantidad de líneas de código para realizar una consulta, inserción, actualización o

eliminación de datos, también permitirá obtener un criterio más real sobre la complejidad de los lenguajes en lo referente a la sintaxis y comprensión.

### **3.4.3 Portabilidad**

Con este parámetro estaremos en la capacidad de determinar cual código (de las tecnologías en estudio) puede ser transferido de una plataforma a otra, cual es la facilidad y cuales son las diferentes dificultades.

### **3.4.5 Interfaz**

Este parámetro nos permitirá determinar cual tecnología presenta a los programadores interfaces de desarrollo mas amigables y con mas facilidades para la realización de aplicaciones tanto Web como de escritorio.

## **3.5 Descripción de los Módulos de Prueba**

### **3.5.1 Modulo 1**

- **Acceso a la Base de Datos:** Este módulo consta de:
  - a.) Métodos de acceso a la base de datos mediante código.
  - b.) La base de datos se llama **consejo** y se encuentra en MySql.

Los métodos de acceso serán desarrollados tanto en Perl como en Python para determinar los puntajes de cada tecnología en el acceso a la base de datos.

### **3.5.2 Módulo 2**

- **Líneas de Código**
  - a.) LDC para Insertar datos en la tabla cantones de la base de datos consejo.

- b.) LDC para actualizar datos en la tabla cantones de la base de datos consejo.
- c.) LDC para eliminar datos en la tabla cantones de la base de datos consejo.
- d.) LDC para consultar datos en la tabla cantones de la base de datos consejo.

Los puntajes serán asignados dependiendo del número de líneas de código utilizado tanto por Perl y Python para realizar estas operaciones fundamentales

### 3.5.3 Módulo 3

- **Portabilidad:** En este módulo se demuestra facilidad de uso de los lenguajes tanto de Perl y Python en las Plataformas Windows y Linux:

Para Windows debemos descargar los módulos respectivos de cada Tecnología.

Para Linux la mayoría de estos módulos vienen incluidos en el SO, simplemente hay que habilitarles.

### 3.5.4 Módulo 4

- **Interfaz de Usuario:** En este módulo indicamos el manejo de los distintos editores de texto para el desarrollo de Aplicaciones Windows/Web tanto en Perl como en Python. Las facilidades que ofrecen cada una de estas herramientas están descritas en la parte correspondiente al desarrollo de este módulo.

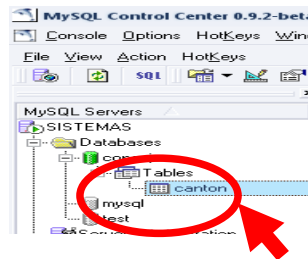
## 3.6 Desarrollo de los Módulos de Prueba

### 3.6.1 Modulo 1 (Acceso a Base de Datos)

#### 3.6.1.1 Modulo1 en Python.

Para la construcción de este módulo utilizaremos acceso a la base de datos por medio de paginas Web.

Para esto primero creamos nuestra base de datos. La base de datos se llama consejo con una tabla llamada cantón, y los campos de la tabla son los siguientes (id\_canton, nom\_canton, hhombres, mujeres, población, analfa\_hombres, analfa\_mujeres, analfabetos, desc\_canton)



**Figura N° III. 9 Base de Datos en Mysql.**

Luego abrimos Dreamweaver y creamos un nuevo sitio, ponemos un nombre (en este caso PYTHON) y luego a continuación creamos una nueva página con extensión .psp, como lo muestra en la figura.



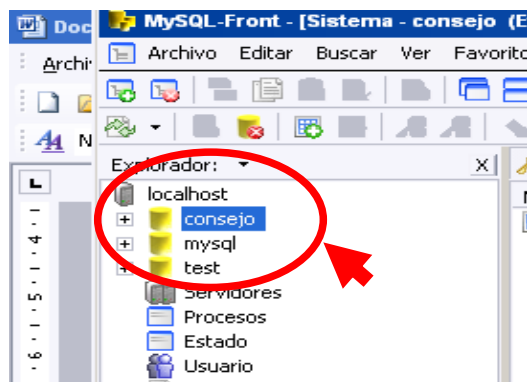
**Figura N° III. 10 Pagina .psp**

Seguidamente en el ambiente para programación de la página ingresamos el código Python para acceder a la base de datos y obtener una vista general de todos los registros existentes en la tabla cantón. Esta página se llamara index.psp.





(id\_canton, nom\_canton, hhombres, mujeres, población, analfa\_hombres, analfa\_mujeres, analfabetos, desc\_canton)



**Figura N° III. 13 Base de Datos para acceder por medio de Perl**

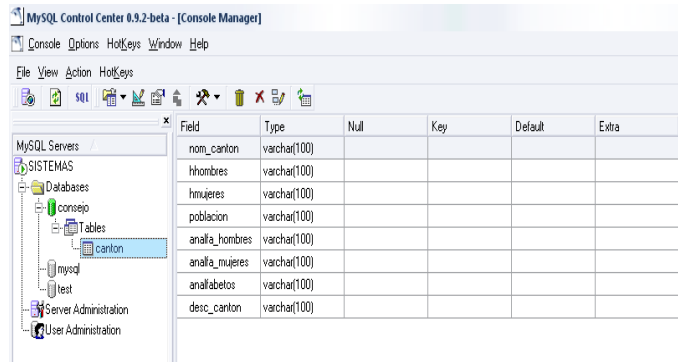
Luego abrimos Dreamweaver y creamos un nuevo sitio, lo ponemos un nombre (en este caso SYSTEM) y luego a continuación creamos una nueva página con extensión .pl, como lo muestra en la figura.



**Figura N° III. 14 Pagina .pl para acceder a la Base de Datos**

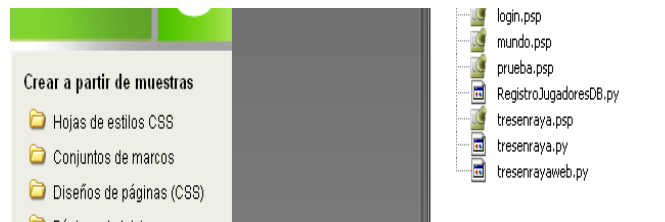
Seguidamente en el ambiente para programación de la página ingresamos el código perl para acceder a la base de datos y obtener una vista general de todos los registros existentes en la tabla cantón. Esta página se llamara index.pl.





**Figura N° III. 17 Base de Datos para el Modulo 2**

Luego abrimos Dreamweaver y creamos una nueva pagina para ingresar el código Python para realizar el ingreso de datos en nuestra tabla cantón de la base de datos consejo.



**Figura N° III. 18 Pagina .psp**

Esta página presentará al usuario una interfaz con los respectivos campos de texto que deberán ser ingresados para su posterior verificación.

Los campos son los siguientes: Nombre Cantón, Habitantes hombres, Habitantes mujeres, Total Habitantes, Analfabetos hombres, Analfabetos mujeres, Total analfabetos.

**Ingreso de Datos**

---

Nombre Canton:

Habitantes Hombres:

Habitantes Mujeres:

Total Habitantes:

Analfabetos Hombres:

Analfabetos Mujeres:

Total Analfabetos:

Descripción:

[Regresar](#)

**Figura N° III. 19 Formulario Ingreso de Datos Cantón**

El código para realizar el ingreso de un nuevo registro en la base de datos se encuentra en Anexos.

Luego ingresamos datos en el formulario. Estos datos serán insertados en la base de datos consejo en la tabla cantón.

**Ingreso de Datos**

---

Nombre Canton:

Habitantes Hombres:

Habitantes Mujeres:

Total Habitantes:

Analfabetos Hombres:

Analfabetos Mujeres:

Total Analfabetos:

Descripción:

[Regresar](#)

**Figura N° III. 20 Datos Ingresados**

Los resultados de la inserción se presentan al usuario en la siguiente página Web.

### Lista de Datos

Canton	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripcion
	Hombres	Mujeres		Hombres	Mujeres		
Alausi	10	10	10	10	10	10	Cantoncito
Riobamba	12	12	24	12	12	24	Canton
Riobamba	200	200	400	300	300	600	Prueba del modulo en python

[Regresar](#)

**Figura N° III. 21 Datos Ingresados en la Base de Datos**

### Actualización y Consulta de Registros.

Para la consulta y actualización de datos lo realizamos mediante una pagina actualización.psp, la misma que contiene la conexión a la base de datos consejo y el código Python para la consulta y actualización de datos. Esta página presentara al usuario la siguiente interfaz:

**Actualizar Datos**

---

Nombre del Canton:

---

[Regresar](#)

**Figura N° III. 22 Formulario para la Actualización de Datos**

Esta pantalla consta de un campo de texto que el usuario debe llenar para indicar el nombre del cantón del cual se van actualizar los datos, luego debe pulsar enviar.

A continuación se le presenta en pantalla los datos del cantón que ingreso anteriormente.

**Datos Canton**

Canton	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripcion
	Hombres	Mujeres		Hombres	Mujeres		
Alausi	10	10	10	10	10	10	Cantoncito

**Actualizacion de Datos**

Nombre Canton:

Habitantes Hombres:

Habitantes Mujeres:

Total Habitantes:

Analfabetos Hombres:

Analfabetos Mujeres:

Total Analfabetos:

Descripcion:

[Regresar](#)

**Figura N° III. 23 Datos Ingresados para la Actualización de Datos**

En esta pantalla el usuario podrá ingresar los nuevos datos para que se almacenen. El código para la actualización de datos se encuentra mas especificado en anexos

### Eliminación de Registros.

Para la eliminación de datos igualmente utilizamos la pagina eliminación.psp en la cual el usuario deberá ingresar el nombre del cantón que desea eliminar. La interfaz presentada al usuario es la siguiente:

**Eliminar Datos**

Nombre del Canton:

[Regresar](#)

**Figura N° III. 24 Formulario para la Eliminación de Datos**

Este formulario consta de un campo de texto donde el usuario ingresara el nombre del cantón a eliminar. El código de la eliminación de datos en Python se encuentra en anexos

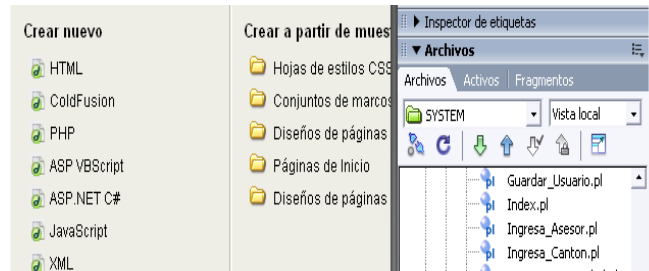
### 3.6.2.1 Modulo 2 en Perl.

Para la construcción de este modulo primero verificamos los campos existentes en nuestra tabla cantón de la base de datos consejo:

Nombre	Tipo	N.	Por d...	Extras
Índice principal	id			unique auto_increment
id	int(11)	SI		
nom_canton	varchar(11)	SI	<NULL>	
hhombres	int(11)	SI	<NULL>	
hmujeres	int(11)	SI	<NULL>	
poblacion	int(11)	SI	<NULL>	
analfa_hombres	int(11)	SI	<NULL>	
analfa_mujeres	int(11)	SI	<NULL>	
desc_canton	int(11)	SI	<NULL>	

**Figura N° III. 25 Campos de la Base de Datos**

A continuación abrimos Dreamweaver y creamos una nueva página para ingresar el código Perl para realizar el ingreso de datos en nuestra tabla cantón de la base de datos consejo.



**Figura N° III. 26 Paginas .pl**

Esta página presentará al usuario una interfaz con los respectivos campos de texto que deberán ser ingresados para su posterior verificación. Los campos son los siguientes:  
(Ver figura N° III. 27)

The image shows a web form titled 'Registro de Datos' (Data Registration). The form has a header 'Datos del Cantón' (Canton Data). It contains five text input fields: 'Nombre Cantón:', 'Habitantes Hombres:', 'Habitantes Mujeres:', 'Analfabetos Hombres:', and 'Analfabetos Mujeres:'. Below these is a larger text area for 'Descripción'. At the bottom of the form are two buttons: 'Aceptar' (Accept) and 'Borrar' (Delete).

**Figura N° III. 27 Formulario para Ingresar Datos en Perl**

El código para realizar el ingreso de un nuevo registro en la base de datos se encuentra en Anexos.

Luego ingresamos datos en el formulario. Estos datos serán insertados en la base de datos consejo en la tabla cantón.

Registro de Datos

Datos del Cantón

Nombre Cantón:

Habitantes Hombres:

Habitantes Mujeres:

Analfabetos Hombres:

Analfabetos Mujeres:

Descripción:

Figura N° III. 28 Datos a ser Insertados

Los resultados de la inserción se presentan al usuario en la siguiente página Web.

Lista de Cantones

Cantón	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripción
	Hombres	Mujeres		Hombres	Mujeres		
Alausi	123	123	246	123	123	246	Cantón
Chambo	2500	1800	4300	594	594	1188	Cantón
Chunchi	1800	1500	3300	552	560	1112	Cantón
Colta	6800	6500	13300	514	578	1092	Cantón
Cumand	3500	1800	5300	347	301	648	Cantón
Guamote	9856	6581	16437	3458	3012	6470	Cantón
Guano	12451	10256	22707	1521	1423	2944	Cantón
Pallatanga	3126	3452	6578	601	100	701	Cantón
Penipe	2451	2561	5012	257	274	531	Cantón
Riobamba	65548	64014	129562	7500	7800	15300	Cantón
Riobamba1	200	200	400	300	300	600	Prueba de Pe

Figura N° III. 29 Datos Ingresados en la Base de Datos

### Actualización y Consulta de Registros.

Para la consulta y actualización de datos lo realizamos mediante una página .pl, la misma que contiene la conexión a la base de datos consejo y el código Perl para la consulta y actualización de datos. Esta página presentará al usuario la siguiente interfaz:

Lista de Cantones

Cantón	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripción	Acción
	Hombres	Mujeres		Hombres	Mujeres			
Alausi	12	12	24	12	12	24	Cantón	
Chambo	2500	1800	4300	594	594	1188	Cantón	
Chunchi	1800	1500	3300	552	560	1112	Cantón	
Colta	6800	6500	13300	514	578	1092	Cantón	
Cumand	3500	1800	5300	347	301	648	Cantón	
Guamote	9856	6581	16437	3458	3012	6470	Cantón	
Guano	12451	10256	22707	1521	1423	2944	Cantón	

Figura N° III. 30 Link para Actualizar Datos

Esta pantalla simplemente hacemos clic sobre la figura del lápiz que se encuentra en la parte derecha de la pantalla y luego se presentará en pantalla los datos para actualizar.



A continuación se le presenta en pantalla los datos del cantón que ingreso anteriormente.

Cantón	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripción
	Hombres	Mujeres		Hombres	Mujeres		
Alausi	12	12	24	12	12	24	Canton

Actualización de Datos:

Nombre del Canton:

Habitantes Hombres:

Habitantes Mujeres:

Analfabetos Hombres:

Analfabetos Mujeres:

Descripcion:

Figura N° III. 31 Datos a ser Actualizados

En esta pantalla el usuario podrá ingresar los nuevos datos para que se almacenen. El código para la actualización de datos se encuentra mas especificado en anexos

### Eliminación de Registros.

Para la eliminación de datos igualmente utilizamos una pagina .pl en la cual el usuario simplemente deberá dar clic sobre la X que se encuentra en la parte derecha de la pantalla y los datos elegidos serán eliminados en forma automática. La interfaz presentada al usuario es la siguiente:

Cantón	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripción	Acciones
	Hombres	Mujeres		Hombres	Mujeres			
Alausi	12	12	24	12	12	24	Canton	
Chambo	2500	1800	4300	594	594	1188	Canton	
Chunchi	1800	1500	3300	552	560	1112	Canton	
Colta	6800	6500	13300	514	578	1092	Canton	
Cumandaz	3500	1800	5300	347	301	648	Canton	
Guamote	9856	6581	16437	3458	3012	6470	Canton	
Guano	12454	10256	22710	1621	1421	2944	Canton	

Figura N° III. 32 Link para Eliminar Datos

El código de la eliminación de datos en Perl se encuentra en anexos

### 3.6.3 Módulo 3 (Portabilidad)

#### 3.6.3.1 Modulo 3 en Python.

Para la construcción de este módulo realizaremos los siguientes pasos:

- Si estamos trabajando en Windows debemos bajar de Internet los módulos de python para instalarlos y configurarlos (Ver la parte de configuración e instalación de Python en Windows).
- Si estamos el Linux Centos 4.3 simplemente verificamos que los módulos hayan sido instalados correctamente puesto que estos ya vienen instalados por defecto.

Para verificar aplicamos los siguientes comandos:

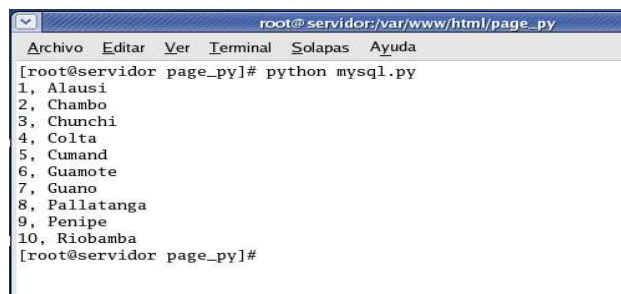
En el shell de Linux digitar `python -V` y luego debemos obtener el siguiente resultado:



```
root@servidor:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@servidor ~]# python -V  
Python 2.3.4  
[root@servidor ~]#
```

**Figura N° III. 33 Modulo Python Instalado en Linux**

Luego digitamos `Python mysql.py` para verificar si tiene conexión a la base de datos y si visualiza los registros. Al realizar este proceso en nuestro caso obtenemos el siguiente resultado.



```
root@servidor:/var/www/html/page_py  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@servidor page_py]# python mysql.py  
1, Alausi  
2, Chambo  
3, Chunchi  
4, Colta  
5, Cumand  
6, Guamote  
7, Guano  
8, Pallatanga  
9, Penipe  
10, Riobamba  
[root@servidor page_py]#
```

**Figura N° III. 34 Acceso a Base de Datos en Python por medio de Consola**

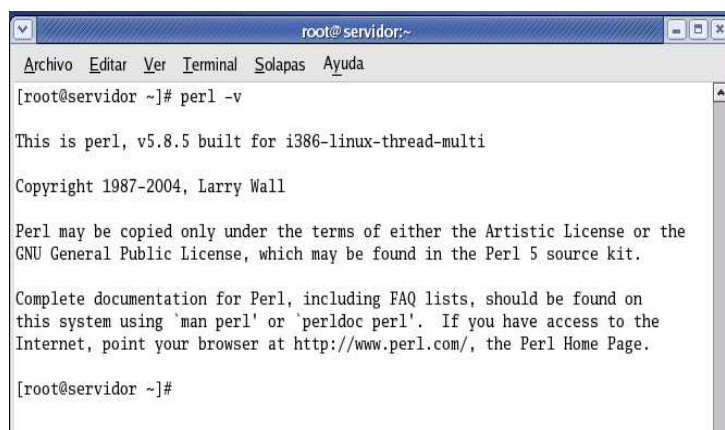
### 3.6.3.2 Modulo 3 en Perl.

Para la construcción de este módulo realizaremos los siguientes pasos:

Si estamos trabajando en Windows debemos bajar de Internet los módulos de Perl para instalarlos y configurarlos (Ver la parte de configuración e instalación de Perl en Windows).

Si estamos el Linux Centos 4.3 simplemente verificamos que los módulos hayan sido instalados correctamente puesto que estos ya vienen instalados por defecto. Para verificar aplicamos los siguientes comandos:

En el shell de Linux digitar `perl -V` y luego debemos obtener el siguiente resultado:



```
root@servidor:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@servidor ~]# perl -v  
  
This is perl, v5.8.5 built for i386-linux-thread-multi  
  
Copyright 1987-2004, Larry Wall  
  
Perl may be copied only under the terms of either the Artistic License or the  
GNU General Public License, which may be found in the Perl 5 source kit.  
  
Complete documentation for Perl, including FAQ lists, should be found on  
this system using 'man perl' or 'perldoc perl'. If you have access to the  
Internet, point your browser at http://www.perl.com/, the Perl Home Page.  
  
[root@servidor ~]#
```

**Figura N° III. 35 Módulos Perl en Linux**

### **3.6.4 Módulo 4 (Interfaz de Desarrollo)**

#### **3.6.4.1 Modulo 4 en Python.**

Para la construcción de este módulo utilizaremos el editor de texto de Dreamweaver y PythonWin los mismos que nos permite desarrollar aplicaciones de escritorio y Web, para lo cual seguimos los siguientes pasos.

Abrimos Dreamweaver y creamos una nueva pagina para ingresar el código python.



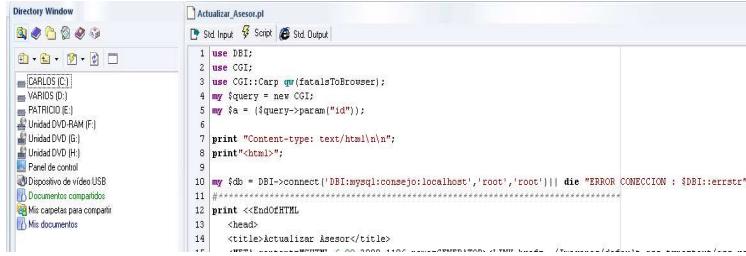


Figura N° III. 38 Interfaz Perl Express

La interfaz que nos presenta para realizar las aplicaciones es la siguiente.

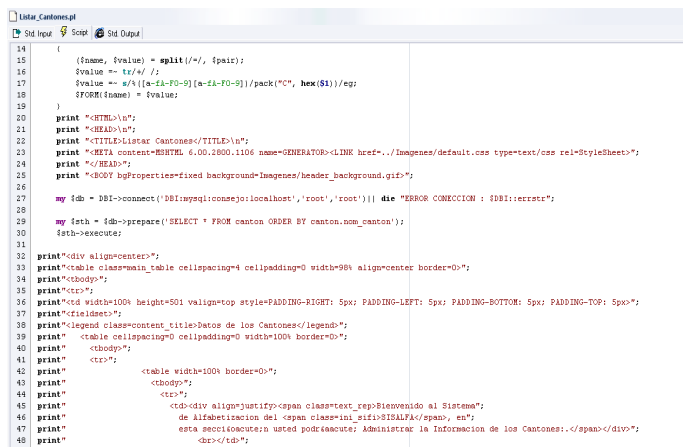


Figura N° III. 39 Interfaz Perl Express

La interfaz que presenta Perl Express es muy amigable para los programadores

### 3.7 Análisis Comparativo

#### 3.7.1 Criterios de Comparación

Los cuatro parámetros generales que hemos tomado en cuenta para desarrollar el estudio comparativo están divididos en varios ítems que detallamos a continuación:

- **Acceso a Base de Datos**
  - **Conexión a la base de datos.**

Permite la conexión a base de datos para su respectiva manipulación.
  - **Manipulación de base de datos.**

Permite tener un control de las tablas, y las consultas que se van a utilizar para la realización de aplicaciones Web dinámicas.
  - **Soporte de base de datos.**

El soporte para varios tipos de base de datos.
  - **Desempeño con base de datos.**

Admite revisar la estructura de los datos y de las tablas sin necesidad de acudir a una herramienta para administración de base de datos.
  
- **Líneas de Código**
  - **Inserción**

Líneas Código utilizado para el ingreso de datos.
  - **Actualización**

Líneas Código utilizado para actualizar datos de una tabla o campo específico de la tabla.
  - **Eliminación**

Líneas Código utilizado para eliminar datos de una tabla.
  - **Consulta**

Líneas Código utilizado para realizar consultas a la base de datos.
  
- **Portabilidad**

- **Plataformas**  
Sea de hardware o software, sobre el cual un programa puede ejecutarse.
- **Versiones**  
Permite corregir las aplicaciones en distintas versiones y que puedan ser interpretadas sin ninguna alteración
- **Máquinas Virtuales**  
Permitir ejecutar varios sistemas operativos simultáneamente sobre el mismo hardware.
- **Compilación**  
Es el proceso por el cual se traducen programas en código fuente a programas en código objeto. El programa que realiza esta traducción se llama compilador.
- **Interfaz de Usuario**
  - **Aplicaciones Web**  
Interfaz visual para el diseño y manipulación de datos.
  - **Aplicaciones Escritorio**  
Interfaz visual para el diseño y manipulación de datos.
  - **Servidor Web**  
Soporte para la publicación de aplicaciones en la Web.
  - **Editores de texto**  
Para la realización de aplicaciones con o sin auto completación de código

La forma para evaluar las dos tecnologías en base a los cuatro criterios antes mencionados y así obtener resultados cuantitativos y cualitativos que permitan una

selección sustentada de una de las tecnologías analizadas, se indican en la siguiente tabla:

**Tabla N°III. 1 Escala de Valoraciones**

<b>Índice</b>	<b>Valor</b>
<b>EA</b>	1
<b>EB</b>	2
<b>EC</b>	3
<b>ED</b>	4

**Tabla N°III. 2 Escala de Valoración Cualitativa**

<b>Valor: 1</b>	<b>Valor: 2</b>	<b>Valor: 3</b>	<b>Valor: 4</b>
Ninguno	Parcialmente	En su mayor parte	Totalmente
No satisfactorio	Poco satisfactorio	Satisfactorio	Muy satisfactorio
Malo	Regular	Bueno	Muy bueno
Inadecuado	Mas o menos	Adecuado	Muy adecuado
Insuficiente	Parcial	Suficiente	Excelente
Deficiente	Poco eficiente	Eficiente	Muy eficiente
Ningún Avance	Cierto avance	Avance significativo	Objetivo Logrado

**Fuente: Proporcionada por el Asesor de Tesis**

Donde:

EA	=	Evaluación Mala
EB	=	Evaluación Regular
EC	=	Evaluación Bueno
ED	=	Evaluación Muy Bueno



**Tabla N°III. 3 Pesos para los Parámetros**

DESCRIPCIÓN	PESO
PV <sub>1</sub>	= [EA, EB, EC, ED]
PV <sub>2</sub>	= [EA, EB, EC, ED]
...	...
PV <sub>i</sub>	= [EA, EB, EC, ED]

Donde:

PV<sub>1</sub>, PV<sub>2</sub>, ..., PV<sub>i</sub> = Peso para las variables en las dos tecnologías en el parámetro.

El valor de cada variable en su respectivo parámetro de interpretación para la tecnología incluye la siguiente fórmula:

$$\begin{array}{lcl} V_1 & \geq & 0 \\ V_2 & \geq & 0 \\ \dots & & \\ V_i & \geq & 0 \end{array} \qquad \begin{array}{lcl} V_1 & \leq & PV_1 \\ V_2 & \leq & PV_2 \\ \dots & & \\ V_i & \leq & PV_i \end{array}$$

Donde:

V<sub>i</sub> = Valor de cada variable en el parámetro.

La calificación definitiva de la herramienta en base a cada parámetro de comparación se obtiene sumando los puntajes obtenidos del análisis, utilizando las siguientes fórmulas:

$$\begin{array}{lcl} P_{pl} & = & \Sigma (V_i) \\ P_{py} & = & \Sigma (V_j) \\ P_c & = & \Sigma (PV_i) \end{array}$$

Donde:

P<sub>pl</sub> = Puntaje acumulado por PERL en el parámetro.

Ppy = Puntaje acumulado por PYTHON en el parámetro.

Pc = Puntaje sobre el que se califica el parámetro.

**Tabla N°III. 4 Porcentajes de los Resultados**

DESCRIPCIÓN	RESULTADO
<b>Cpl</b>	$= (Ppl / Pc) * 100\%$
<b>Cpy</b>	$= (Ppy / Pc) * 100\%$

Donde:

**Cpl:** Calificación que obtuvo la Tecnología PERL en el parámetro

**Cpy:** Calificación que obtuvo la Tecnología PYTHON en el parámetro

**Tabla N°III. 5 Calificación de Tecnologías**

DESCRIPCIÓN	CALIFICACIÓN
<b>Malo</b>	$< 50\%$
<b>Regular</b>	$\geq 50\% \text{ y } < 60\%$
<b>Bueno</b>	$> 60\% \text{ y } < 75\%$
<b>Muy bueno</b>	$\geq 75\% \text{ y } < 90\%$
<b>Excelente</b>	$\geq 90\%$

### **3.7.2 Análisis de los Criterios de Comparación.**

Los valores para este análisis serán tomados de la tabla de Escala de valoración cualitativa. Es decir que estos valores tendrán un máximo de 4 puntos en caso de cumplir excelentemente con la variable del parámetro.

#### **3.7.2.1 Acceso a Base de Datos**

**Tabla N°III. 6 Acceso a Base de Datos Tecnología Perl**

<b>N°</b>	<b>VARIABLES</b>	<b>PERL</b>
<b>1</b>	<b>Conexión a la base de datos.</b>	El proceso de conexión es Satisfactorio.
<b>2</b>	<b>Manipulación de base de datos.</b>	Es excelente porque permite insertar, modificar, consultar y actualizar, datos de la base de datos
<b>3</b>	<b>Soporte de base de datos.</b>	Este lenguaje es bueno para base de datos como: MySQL – PostgreSQL - SqlServer
<b>4</b>	<b>Desempeño con base de datos.</b>	Es excelente.

**Tabla N°III. 7 Acceso a Base de Datos Tecnología Python**

<b>N°</b>	<b>VARIABLES</b>	<b>PYTHON</b>
<b>1</b>	<b>Conexión a la base de datos.</b>	El proceso de conexión es satisfactorio.
<b>2</b>	<b>Manipulación de base de datos.</b>	Es excelente porque permite insertar, modificar, consultar y actualizar, datos de la base de datos
<b>3</b>	<b>Soporte de base de datos.</b>	Este lenguaje es muy bueno para base de datos como: MySQL – PostgreSQL – Sqlite - SqlServer
<b>4</b>	<b>Desempeño con base de datos.</b>	Es excelente

**Fuente: Ver modulo I y Anexo B (Conexión a la base de datos)**

**a. Interpretación de Resultados**

$$\begin{aligned}
 Ppl &= \Sigma (V_i) \\
 Ppl &= \Sigma (3 + 4 + 3 + 4) \\
 Ppl &= 14 \\
 Ppy &= \Sigma (V_j) \\
 Ppy &= \Sigma (3 + 4 + 4 + 4) \\
 Ppy &= 15 \\
 Pc &= \Sigma (PV_i) \\
 Pc &= \Sigma (4 + 4 + 4 + 4) \\
 Pc &= 16 \\
 Cpl &= (Ppl / Pc) * 100\% \\
 Cpl &= (14 / 16) * 100\% \\
 Cpl &= 87.5\% \\
 Cpy &= (Ppy / Pc) * 100\% \\
 Cpy &= (15 / 16) * 100\% \\
 Cpy &= 93.7\%
 \end{aligned}$$

**Tabla N° III. 8 Resultados Parámetros Acceso a Base de Datos**

VARIABLES	TECNOLOGÍAS			
	PERL		PYTHON	
	V <sub>i</sub>	Cpl	V <sub>i</sub>	Cpy
V <sub>1</sub>	3	87.5%	3	93.7%
V <sub>2</sub>	4		4	
V <sub>3</sub>	3		4	
V <sub>4</sub>	4		4	
<b>TOTAL</b>	<b>14</b>		<b>15</b>	

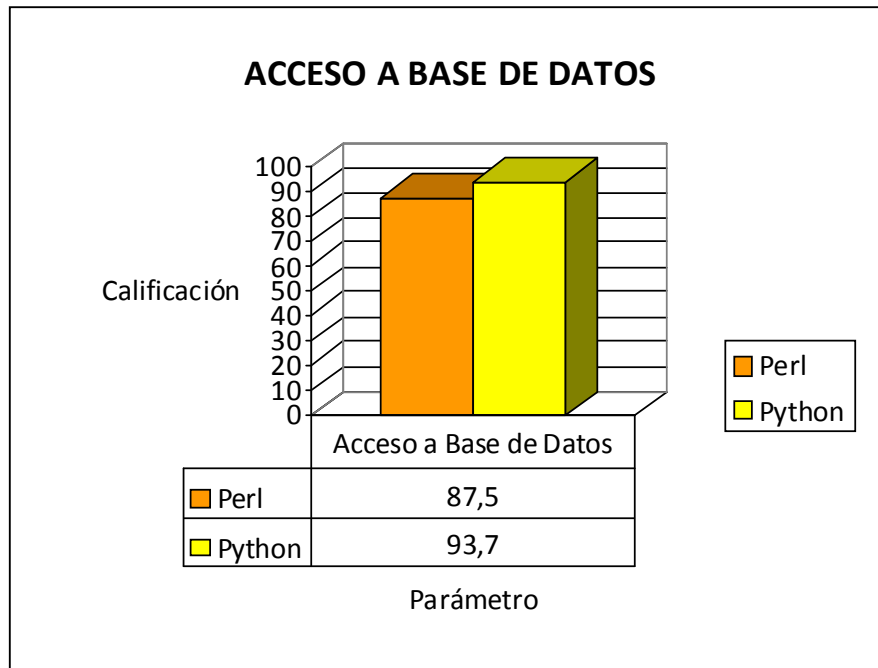


Figura N° III. 40 Gráfico resultado del Parámetro Acceso a Base de Datos

#### b. Descripción de Resultados

PERL es una tecnología que en los últimos tiempos ha ido creciendo y recuperando la popularidad que tenía en años anteriores ya que presenta facilidades para el desarrollo de aplicaciones Web con conexiones a base de datos en forma rápida, comprensible y con pocas líneas de código. Panorama que es igual cuando se trabaja con Python puesto que también el acceso a base de datos se lo realiza en forma rápida, en este caso las dos tecnologías presentan un proceso satisfactorio en la conexión a Base de Datos.

La manipulación de la base de datos en ambas tecnologías es excelente debido a que dichas tecnologías tienen las sentencias SQL, las mismas que son conocidas por los programadores por la cual su manipulación la hace simple para su respectivo desarrollo.

Además las dos tecnologías brindan soporte para varios tipos de base de datos, aunque debemos anotar aquí que Python presenta ciertas facilidades principalmente en el código para la conexión a otras bases de datos con relación a Perl

Las 2 tecnologías presentan un desempeño excelente con base de datos

### 3.7.2.2 Líneas de Código

Tabla N°III. 9 Líneas de Código Tecnología Perl

N°	VARIABLES	PERL
1	Inserción	Totalmente simple
2	Actualización	Totalmente simple
3	Eliminación	Totalmente simple
4	Consultas	Totalmente simple

Tabla N°III. 10 Líneas de Código tecnología Python

N°	VARIABLES	PYTHON
1	Inserción	En su mayor parte, Simple.
2	Actualización	Totalmente Simple.
3	Eliminación	Totalmente Simple.
4	Consultas	En su mayor parte Simple.

Fuente: Ver Modulo II y Anexo B (Operaciones Básicas en Perl y Python)

#### a. Interpretación de Resultados

$$Ppl = \Sigma (V_i)$$

$$Ppl = \Sigma (4 + 4 + 4 + 4)$$

$$\begin{aligned}
 P_{pl} &= 16 \\
 P_{py} &= \Sigma (V_j) \\
 P_{py} &= \Sigma (3 + 4 + 4 + 3) \\
 P_{py} &= 14 \\
 P_c &= \Sigma (P_{V_i}) \\
 P_c &= \Sigma (4 + 4 + 4 + 4) \\
 P_c &= 16 \\
 C_{pl} &= (P_{pl} / P_c) * 100\% \\
 C_{pl} &= (16 / 16) * 100\% \\
 C_{pl} &= 100\% \\
 C_{py} &= (P_{py} / P_c) * 100\% \\
 C_{py} &= (14 / 16) * 100\% \\
 C_{py} &= 87.5\%
 \end{aligned}$$

Tabla N°III. 11 Resultados Parámetro Líneas de Código

VARIABLES	TECNOLOGÍAS			
	PERL		PYTHON	
	PV <sub>i</sub>	Cpl	PV <sub>i</sub>	Cpy
V <sub>1</sub>	4	100%	3	87.5%
V <sub>2</sub>	4		4	
V <sub>3</sub>	4		4	
V <sub>4</sub>	4		3	
TOTAL	16		14	

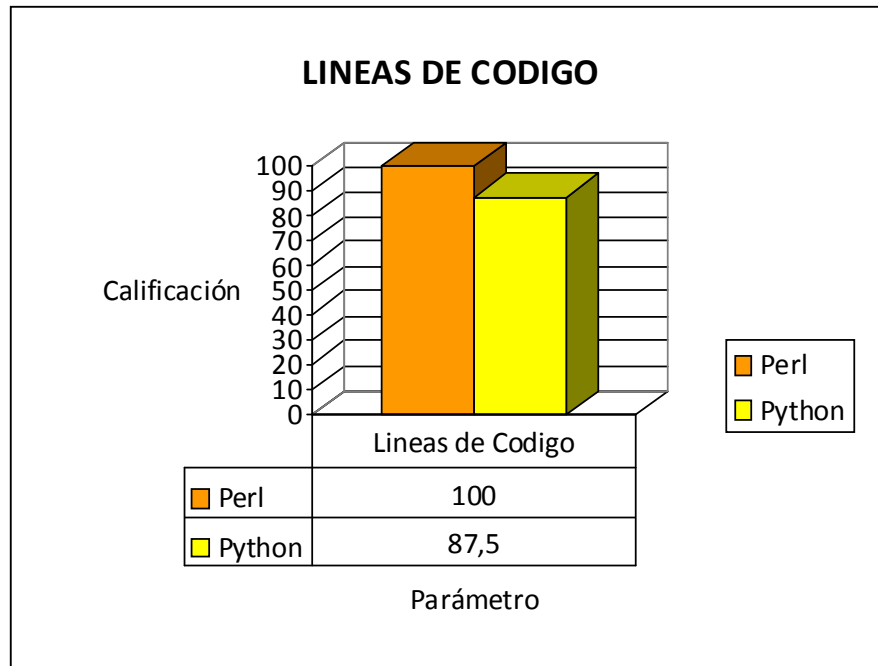


Figura N° III. 41 Grafico resultado del Parámetro Líneas de Código

### b. Descripción de Resultados

En Perl realizar las operaciones de: Inserción, actualización, eliminación y consulta es totalmente simple, con sintaxis comprensible para este caso. En Python son totalmente simples las operaciones como actualización y eliminación de datos pero las operaciones como inserción y consulta son en su mayor parte simples debido a que utiliza un número mayor de líneas de código. Esta diferencia no es tan evidente ya que en ocasiones la diferencia es una línea de código.



### 3.7.2.3 Portabilidad

**Tabla N°III. 12 Parámetro Potabilidad Tecnología PERL**

N°	VARIABLES	PERL
1	Plataformas	Es muy adecuado para sistemas operativos como: Linux, FreeBSD, Windows XP, MacOS, DOS, Novell y otros
2	Versiones	Totalmente probable que las aplicaciones corran en distintas versiones.
3	Máquinas Virtuales	Muy adecuado, porque trabaja como una máquina física.
4	Compilación	Necesita totalmente de compilación para su respectiva ejecución.

**Tabla N°III. 13 Parámetro Portabilidad Tecnología PYTHON**

N°	VARIABLES	PYTHON
1	Plataformas	Es muy adecuado para sistemas operativos como: Linux, Unix, Windows, MacOS y otros
2	Versiones	En su mayor parte es probable que las aplicaciones corran en distintas versiones.
3	Máquinas Virtuales	Muy adecuado, porque trabaja como una máquina física.
4	Compilación	Necesita totalmente de compilación para su respectiva ejecución.

**Fuente:** Ver Modulo III Portabilidad

#### a. Interpretación de Resultados

$$Ppl = \Sigma (V_i)$$

$$Ppl = \Sigma (4 + 4 + 4 + 4)$$

$$\begin{aligned}
 P_{pl} &= 16 \\
 P_{py} &= \Sigma (V_i) \\
 P_{py} &= \Sigma (4 + 3 + 4 + 4) \\
 P_{py} &= 15 \\
 P_c &= \Sigma (P_{V_i}) \\
 P_c &= \Sigma (4 + 4 + 4 + 4) \\
 P_c &= 16 \\
 C_{pl} &= (P_{pl} / P_c) * 100\% \\
 C_{pl} &= (16 / 16) * 100\% \\
 C_{pl} &= 100 \% \\
 C_{py} &= (P_{py} / P_c) * 100\% \\
 C_{py} &= (15 / 16) * 100\% \\
 C_{py} &= 93.7 \%
 \end{aligned}$$

**Tabla N°III. 14 Resultado Parámetro Portabilidad**

VARIABLES	TECNOLOGÍAS			
	PERL		PYTHON	
	PV <sub>i</sub>	Cpl	PV <sub>i</sub>	Cpy
V <sub>1</sub>	4	100 %	4	93.7 %
V <sub>2</sub>	4		3	
V <sub>3</sub>	4		4	
V <sub>4</sub>	4		4	
<b>TOTAL</b>	<b>15</b>		<b>15</b>	

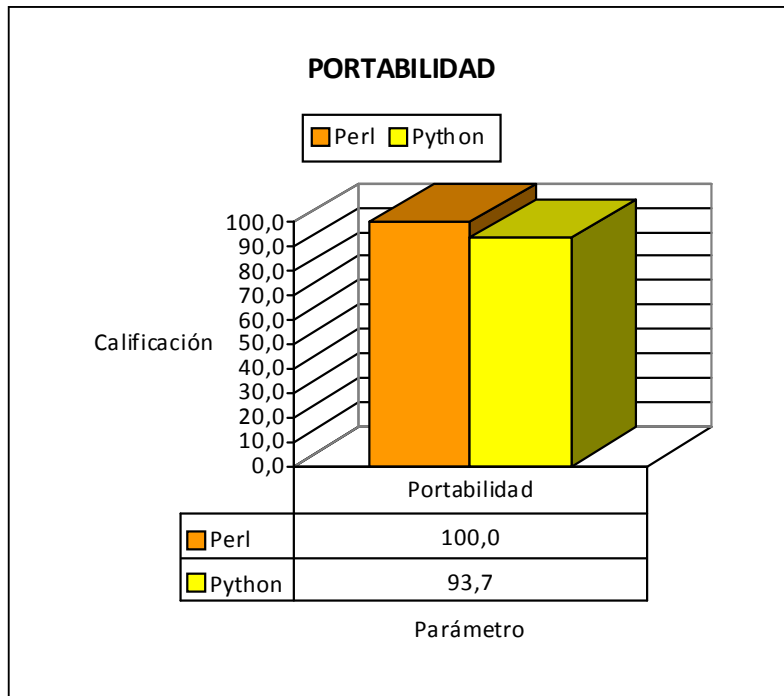


Figura N° III. 42 Gráfico de Resultado de Portabilidad

### b. Descripción de Resultados

Tanto Perl y Python son portables entre sistemas operativos toda vez que los módulos de estas tecnologías o ya vienen instalados en los sistemas operativos o son de fácil adquisición.

Con relación al funcionamiento entre versiones es totalmente probable que Perl funcione en distintas versiones, en cambio Python es probable que en su mayor parte que no presente ningún inconveniente ya que ambos lenguajes son OpenSource y se puede encontrar información fácilmente.

Ambas tecnologías son muy adecuadas y pueden ejecutarse sin ningún problema en máquinas virtuales.

Tanto la tecnología Perl y Python necesitan totalmente compilarse para su ejecución en este caso.

### 3.7.2.4 Interfaz de Usuario

Tabla N°III. 15 Parámetro Interfaz Tecnología PERL

N°	VARIABLES	PERL
1	Aplicaciones Web	Tiene avance significativo con Macromedia. Visual .Net, Perl Express 5
2	Aplicaciones Escritorio	Existe avance significativo
3	Servidor Web	Trabaja excelente con Apache, Ironwal Xampp, Stunnix, IIS.
4	Editores de Texto	Muy bueno con Varios editores

Tabla N°III. 16 Parámetro Interfaz Tecnología PYTHON

N°	VARIABLES	PYTHON
1	Aplicaciones Web	Tiene avance significativo con Macromedia, PythonWin
2	Aplicaciones Escritorio	Existe avance significativo
3	Servidor Web	Trabaja excelente con Apache, Wamp, IIS.
4	Editores de Texto	Muy bueno con Varios editores

Fuente: Ver Modulo IV Interfaz de Usuario

#### a. Interpretación de Resultados

$$Ppl = \Sigma (V_i)$$

$$Ppl = \Sigma (3 + 3 + 4 + 4)$$

$$\begin{aligned}
 P_{pl} &= 14 \\
 P_{py} &= \Sigma (V_i) \\
 P_{py} &= \Sigma (3 + 3 + 4 + 4) \\
 P_{py} &= 14 \\
 P_c &= \Sigma (P V_i) \\
 P_c &= \Sigma (4 + 4 + 4 + 4) \\
 P_c &= 16 \\
 C_{pl} &= (P_{pl} / P_c) * 100\% \\
 C_{pl} &= (14 / 16) * 100\% \\
 C_{pl} &= 87.5 \% \\
 C_{py} &= (P_{py} / P_c) * 100\% \\
 C_{py} &= (14 / 16) * 100\% \\
 C_{py} &= 87.5 \%
 \end{aligned}$$

**Tabla N°III. 17 Resultado Parámetro Interfaz**

VARIABLES	TECNOLOGÍAS			
	PERL		PYTHON	
	PV <sub>i</sub>	Cpl	PV <sub>i</sub>	Cpy
V <sub>1</sub>	3	87.5 %	3	87.5 %
V <sub>2</sub>	3		3	
V <sub>3</sub>	4		4	
V <sub>4</sub>	4		4	
<b>TOTAL</b>	<b>14</b>		<b>14</b>	

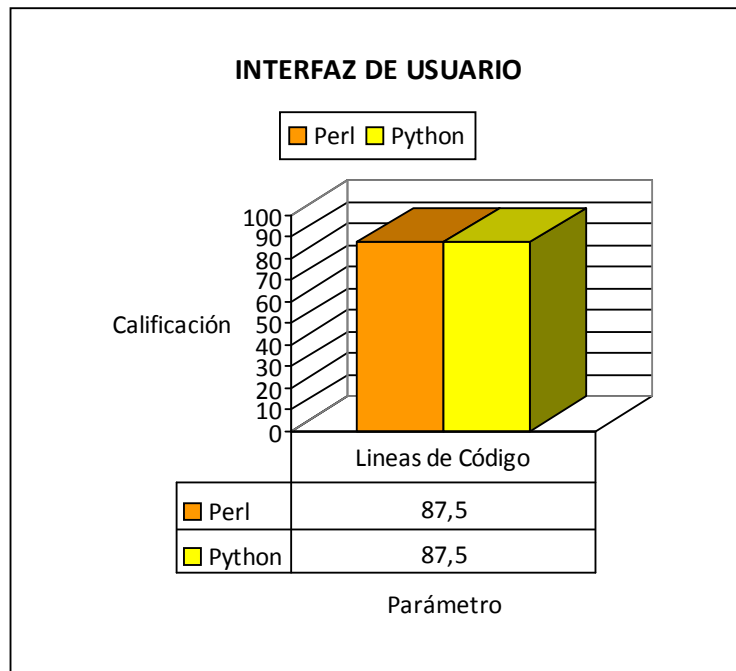


Figura N° III. 43 Gráfico Resultado Parámetro Interfaz

### b. Descripción de Resultados

Tanto Perl como Python tienen interfaces de desarrollo para programadores Web, aunque algunas interfaces son más amigables que otras.

Dentro del desarrollo de las aplicaciones de escritorio en las dos tecnologías existe un avance significativo ya que por ejemplo Macromedia Dreamweaver permite desarrollar aplicaciones de escritorio tanto en Perl como en Python.

Ambas tecnologías trabajan excelentemente bien con los servidores como es el caso del servidor apache para la publicación de las aplicaciones en la Web.

Así mismo las dos tecnologías trabajan muy bien con varios editores de texto para la creación de aplicaciones.

### 3.7.3. Puntajes Alcanzados

El puntaje final y el porcentaje que ha obtenido cada tecnología de obtiene de la siguiente manera:

**Puntaje Total del Análisis**

$$PT = \Sigma (Pc)$$

**Puntaje Total de Perl**

$$PTpl = ((\Sigma(Ppl)) / PT) * 100\%$$

**Puntaje Total de Python**

$$PTpy = ((\Sigma(Ppy)) / PT) * 100\%$$

**Tabla N°III. 18 Tabla General de Resultados**

PARÁMETRO	VARIABLES	TECNOLOGÍAS	
		PERL	PYTHON
Acceso a Base de Datos	Conexión a la base de datos.	3	3
	Manipulación de base de datos.	4	4
	Soporte de base de datos.	3	4
	Desempeño con base de datos.	4	4
Líneas de Código	Inserción	4	3
	Actualización	4	4
	Eliminación	4	4
	Consulta	4	3
Portabilidad	Plataformas	4	4
	Versiones	4	3
	Máquinas Virtuales	4	4
	Compilación	4	4
Interfaz de Usuario	Aplicaciones Web	3	3
	Aplicaciones Escritorio	3	3
	Servidor Web	4	4
	Editores de texto	4	4
<b>TOTAL</b>		<b>60</b>	<b>58</b>

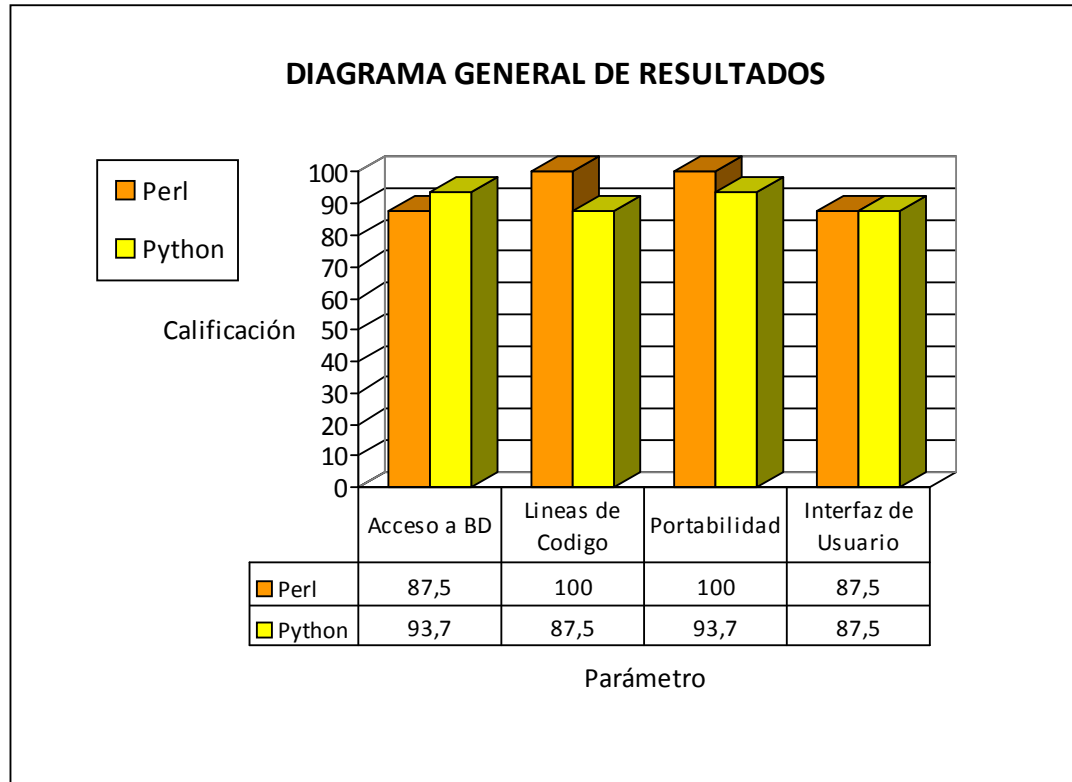


Figura N° III. 44 Gráfico General de Resultados

**Puntaje Total del Análisis**

$$PT = \Sigma (Pc)$$

$$PT = 64$$

**Puntaje Total de PERL**

$$PTpl = ((\Sigma(Ppl)) / PT) * 100\%$$

$$PTpl = (60 / 64) * 100\%$$

$$PTpl = 93.75 \%$$

**Puntaje Total de PYTHON**

$$PTpy = ((\Sigma(Ppy)) / PT) * 100\%$$



$$PTpy = (58/64) * 100\%$$

$$PTpy = 90.62 \%$$

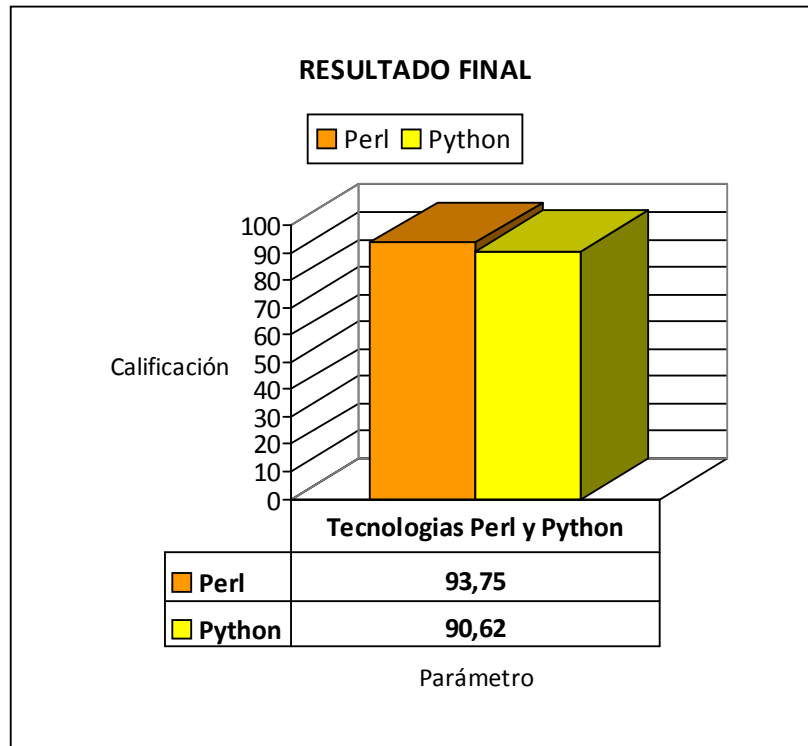


Figura N° III. 45 Gráfico Resultado Final

### 3.7.3.1 Interpretación de Resultados.

Como podemos observar en el resultado final, las dos tecnologías son excelentes para el desarrollo de aplicaciones Web. Como las 2 se encuentran en constante evolución presentan facilidades de uso y de manipulación para los desarrolladores.

La pequeña diferencia que existe se debe principalmente a que en las operaciones de inserción y consulta de datos, Python tiene una línea más de código que Perl para realizar estas operaciones, esta diferencia no es tan evidente cuando se realizan

aplicaciones grandes. Python recompensa esta diferencia con su sintaxis que es más elegante y entendible. También existe una diferencia marcada en lo referente a la portabilidad, ya que al ser Perl el lenguaje que sirvió de punto de partida para el apareamiento de Python, es totalmente probable que funcione en distintas versiones.

No podemos anotar que las ventajas de Perl son desventajas de Python ni viceversa, puesto que las dos tecnologías en su mayor parte tienen las mismas ventajas y desventajas, es por esto que algunos autores de artículos sobre estos lenguajes les han denominado los lenguajes primos hermanos. Es decir que podemos desarrollar una misma aplicación tanto en Perl como en Python con las mismas características y la misma funcionalidad. Pero debemos anotar que Perl es el lenguaje que ha servido de inicio para la creación de otros lenguajes de desarrollo como Python, Java, Jython etc. Así mismo Perl ha servido como lenguaje crítico ante las falencias de los otros lenguajes, ya que siempre ha estado en constante cambio y además ha seguido en búsqueda de mejorar su potencial y popularidad bien ganada desde tiempos atrás. Por el resultado obtenido, podemos observar que Perl es el ganador con una pequeña diferencia, mencionado en otras palabras diremos que entre los dos lenguajes existe un empate técnico.

### 3.7.4 Resultados del Análisis.

Luego del análisis correspondiente realizado a las Tecnologías Perl y Python en el desarrollo de aplicaciones Web, podemos determinar lo siguiente:

- La conexión a base de datos en las dos tecnologías es bastante simple ya que presentan facilidades para el desarrollo de distintas aplicaciones. Estas facilidades principalmente tienen que ver con que ambas tecnologías presentan a los desarrolladores, sintaxis comprensibles para la conexión a base de datos y con pocas líneas de código.
- La manipulación de base de datos en ambas tecnologías es excelente debido a que tienen las sentencias Sql las mismas que son conocidas por los programadores.
- Además estas tecnologías brindan soporte para varios tipos de base de datos, aunque Python brinda facilidades en el código para la conexión a otras bases de datos.
- En Perl las operaciones funcionales como: Inserción, actualización, eliminación y consulta de datos es bastante simple. En Python estas mismas operaciones son también bastante simples sino que las realiza con un numero mayor de líneas de código. Esta diferencia no es tan evidente ya que en ocasiones se diferencia en una línea de código.
- Tanto Perl y Python son portables entre sistemas operativos toda vez que los módulos de estas tecnologías ya vienen instalados en los sistemas operativos o son de fácil adquisición.
- Como las dos tecnologías son OpenSource podemos determinar que el funcionamiento de sus aplicaciones entre varias versiones probablemente no

presente ningún inconveniente ya que son lenguaje libre y se puede encontrar información fácilmente.

- Con relación a lo que tiene que ver con Interfaz de desarrollo, tanto Perl como Python presentan interfaces amigables para los desarrolladores. Debemos anotar que cada tecnología se encuentra en constante evolución ya que en cada una de sus comunidades sus seguidores se encuentran trabajando para presentar editores más amigables y de fácil desarrollo para los programadores, como es el caso de Python.net

**Conclusión:**

Por lo anteriormente anotado, y luego obtener de los resultados del análisis comparativo llegamos a la conclusión que la tecnología Perl es la más adecuada para el desarrollo de nuestra aplicación Web, puesto que ha tenido un mejor desenvolvimiento en los diferentes parámetros descritos en este trabajo de Investigación, principalmente en los parámetros como líneas de código y portabilidad, en los cuales Perl ejerce un amplio dominio con relación a Python.

## **CAPITULO IV**

### **DESARROLLO DEL SISTEMA SISALFA PARA EL PROGRAMA DE ALFABETIZACION “YO SI PUEDO”.**

#### **Fase 1:**

#### **PREPROYECTO**

##### **4.1 Identificación del Proyecto**

El propósito de la creación de la presente aplicación es ayudar a obtener reportes estadísticos sobre la marcha del programa de alfabetización de la provincia.

##### **4.1.1 Financiamiento y Comisión del proyecto**

Dado que el proyecto se lo realizara como en código libre no tendrá costo alguno ni comisión alguna en lo que se refiere al Desarrollo.

La especificación de los puntos anteriores evitará problemas en fases posteriores del proyecto.

##### **4.1.2 Reglas**

1. **Must have:** Debe tener. Esencialmente

- Modulo 1. Administrador (Seguridades)
  
- Módulo 2. Reportes
  - Control de Matriculas de los alfabetizados (Matriculas ordinarias y extraordinarias)
  
  - Reportes estadísticos por cantón, parroquia y comunidad sobre la marcha del programa de alfabetización.
  
  - Reportes de los asesores y educadores de cada cantón
  
  - Reportes del presupuesto que invierte el consejo provincial en cada cantón para emprender el programa.

2. **Should have:** Debería tener.

- Reporte mas detallados sobre los alumnos en cada cantón.

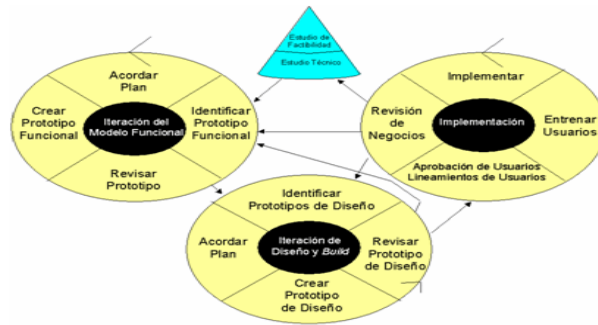
3. **Could have:** Podría tener.

- Información sobre los programas de alfabetización en las otras provincias.

4. **Want to have but won't have this time around:** Se desea que tenga, pero no lo tendrá. No podemos describir este requerimiento dado que se cumplirá lo requerido para el proyecto.

**Fase 2:**

## ETAPAS VITALES DEL PROYECTO



FiguraNº IV 1 Etapas Vitales del Proyecto

La descripción de proceso en la figura anterior demuestra el ciclo vital del proyecto de esta fase de nuestra metodología DSDM. Representa las 5 etapas que tendrá nuestro proyecto. Las primeras dos etapas, el estudio de factibilidad y el estudio Técnico son las fases secuenciales que se complementan. Al concluirse estas fases, el sistema se desarrolla iterativo e incremental en las etapas de modelo, diseño e iteración y de la puesta en práctica funcionales de la estructura.

### 4.2 Etapa 1: El Estudio De Viabilidad

#### 4.2.1 Requisitos previos para usar DSDM (Dynamic Systems Development Method)

Para tener éxito con la utilización de la metodología DSDM analicemos los requisitos previos que necesitan ser comprendidos.

**1.- Interoperablemente activo** Entre el equipo del proyecto, usuarios, terminales futuros y dirección más alta.

**2.- Descomposición del proyecto.** La posibilidad de descomposición en las partes más pequeñas habilita el acercamiento reiterativo, y actividades que son difíciles de priorizar y causan a menudo retrasos.

Ya que DSDM fue desarrollado justamente para cubrir con estos dos requisitos vemos conveniente la utilización de esta metodología para nuestro proyecto a desarrollar.

### **Riesgos más importantes implicados**

Uno de los riesgos para el que DSDM no está bien preparado para lo que tiene que ver con seguridades-críticas. La comprobación extensa y aprobación encontradas en estos tipos de proyectos chocan a tiempo con metas de DSDM en presupuesto. Finalmente, no podrían satisfacerse proyectos que apuntan a los componentes re-utilizables para desarrollo que usa DSDM.

## **4.2.2 INGENIERIA DE LA INFORMACIÓN**

### **4.2.2.1 Definición del Ámbito**

En la actualidad el departamento de Gestión Social del H. Consejo Provincial se encuentra encargado de la ejecución del programa de alfabetización dentro de la provincia.

Este departamento no cuenta con un sistema informático que automatice este proceso por lo que los reportes se lo elaboran en documentos de Excel, impidiendo de esta manera que un usuario común pueda obtener estos reportes, puesto que esta información solamente la poseen los asesores o coordinadores de cada cantón, es decir que si se requiere un informe del programa de alfabetización este debe esperar hasta la llegada de los asesores cada fin de semana.



#### **4.2.2.2 Requerimientos**

Para solucionar los inconvenientes citados en el literal anterior se puede desarrollar un sistema, el mismo que permita:

- Ingresar, Consultar y Actualizar la información acerca de los estudiantes matriculados en el programa de alfabetización en cada cantón, parroquia o comunidad.
- Ingresar, Consultar y Actualizar la información acerca de los asesores, educadores, coordinadores del programa en cada cantón.
- Ingresar, Consultar y Actualizar la información acerca del presupuesto que cada cantón recibe para la ejecución del programa.
- Generar reportes estadísticos de la marcha del programa en los cantones, parroquias, y comunidades.

#### **4.2.2.3 Estudio de Factibilidad**

##### **4.2.2.3.1 Factibilidad Económica**

Como el sistema será desarrollado en el lenguaje de código libre Perl, la institución no tendrá que invertir en la compra de licencias de productos. Además los recursos hardware y Software del consejo provincial son plenamente aptos para la ejecución y desarrollo de la aplicación Web.

##### **4.2.2.3.2 Factibilidad Técnica**

###### **a. Recurso Humano**

El recurso humano con el que se cuenta para el desarrollo del sistema es:

- Ing. Danilo Pastor Director
- Ing. Roberto Insuasti Miembro del Tribunal
- Ing. Jaime Zarate Jefe(E) de la unidad de Sistemas
- Dr. Napoleón Pino Jefe del departamento de Gestión Social.
- Carlos Chávez Desarrollador
- Iván Buñay Desarrollador

#### **b. Recursos Software**

**Sistema Operativo:** Linux Centos 4.3

**Servidor de Base de Datos:** Mysql 4.1

**Servidor Web:** Apache Tomcat 5.0

**Herramienta de desarrollo:** Active Perl 5.8.x.

#### **4.2.2.3.3 Factibilidad Operativa**

Para desarrollar este sistema se cuenta con el apoyo total del recurso humano mencionado anteriormente.

#### **4.2.2.3.4 Factibilidad Legal**

Existe la autorización de las autoridades respectivas por lo que no existe ningún tipo de impedimento legal para el desarrollo del sistema.

#### **4.2.2.4 Planteamiento de la solución**

Nuestro sistema tendrá los siguientes módulos:

### **1.- Modulo Administrador:**

Este modulo tendrá los privilegios de:

- Crear cuentas de usuario administrador
- Ingreso, actualizar, listar, estudiantes, matriculas, asesores, educadores, etc.
- Obtener reportes acerca de la ejecución del programa, tanto gráficos como en porcentajes
- Ingreso, actualización, del presupuesto asignado a cada cantón para la ejecución del programa.

### **2.- Modulo Limitado:**

Este modulo tendrá los privilegios de:

- Ingreso, actualizar, listar, estudiantes, matriculas, asesores, educadores, etc.
- Obtener reportes acerca de la ejecución del programa, tanto gráficos como en porcentajes
- Ingreso, actualización, del presupuesto asignado a cada cantón para la ejecución del programa.

### **3.- Modulo Usuario:**

Este modulo tendrá los privilegios de:

- Obtener reportes acerca de la ejecución del programa, tanto gráficos como en porcentajes

#### **4.2.2.5 Planificación Temporal**

Ver Anexo A

#### **4.2.2.6 Especificación de Requerimientos (SRS)**

Ver Anexo A

### **4.3 Etapa 2:**

#### **ANÁLISIS DEL SISTEMA.**

##### **4.3.1 Casos de Uso del Sistema.**

La técnica que permite mejorar el entendimiento de los requerimientos es la descripción de los casos de uso. Los casos de uso son una breve descripción de la utilización del sistema; no son exactamente los requerimientos ni las especificaciones funcionales, sino que ejemplifica e incluyen tácitamente los requerimientos del sistema.

Se han identificado los siguientes casos de uso para el desarrollo del Sistema

##### **4.3.1.1 Casos de Usos de Alto nivel Administrador**

###### **CASOS DE USO #1**

**NOMBRE:** Gestión de cuentas (Inicios Sesión)

**CÓDIGO:** CUGC

**ACTORES:** Administrador

**TIPO:** Primario

**DESCRIPCIÓN:** El Administrador, bajo pedido de usuario realiza la creación o modificación de una cuenta o inicio de sesión.

En la gestión se realiza dos procesos: el primero se encarga de generar cuentas de usuario en un determinado grupo (administrador) asignación de cuentas y contraseñas.

Y el segundo proceso se encarga de la modificación de los parámetros que intervienen dentro de una cuenta de usuario.

##### **4.3.1.2 Casos de Uso de alto nivel Limitada**

## CASO DE USO #2

**NOMBRE:** Limitada

**Código:** CUAD

**ACTORES:** Limitada

**TIPO:** Primario

**DESCRIPCIÓN:** El usuario limitado ingresará al sistema luego de autenticarse. Una vez autenticado podrá acceder a la información existente de dicho sistema, reportes, así como también podrá realizar el ingreso, actualización, de los diferentes datos que existen en el sistema.

### 4.3.1.3 Casos de Uso de Alto nivel Usuario

## CASOS DE USO #3

**NOMBRE:** Obtención de Reportes

**Código:** CUS

**ACTORES:** Sistema

**TIPO:** Primario, usuario

**DESCRIPCIÓN:** El Sistema es el encargado de generar los respectivos reportes previa petición de los usuarios.

### 4.3 Etapa 3: Iteración Modelo Funcional



FiguraNº IV 2 Modelo Funcional

Los requisitos que se han identificado en las etapas anteriores se convierten a un modelo funcional. Este modelo consiste en un prototipo de funcionamiento y modelos. Prototyping es una de las técnicas dominantes del proyecto dentro de esta etapa que ayude a realizar la buena implicación del usuario a través del proyecto. El prototipo desarrollado es repasado por los diversos grupos de usuario.

#### **4.3.1 Análisis**

Con lo analizado en la Fase anterior continuaremos con la fase de Análisis, en esta fase se definiremos el problema con los conceptos relacionados en los casos de uso que estamos tratando.

##### **4.3.1.1 Definir y Refinar los casos de uso**

Los Casos de Uso nos ayudarán a describir la secuencia de eventos de un actor cuando utiliza un sistema. En sí es una forma particular de usar un sistema para ilustrar e implicar los requisitos.

Consideraremos los más importantes en el formato expandido.

#### **Sistema Propuesto**

##### **Casos de Uso de alto nivel Administrador**

##### **CASO DE USO #1**

**NOMBRE:** Gestión de cuentas (Inicio Sesión)

**Código:** CUGC

**ACTORES:** Administrador

**TIPO:** Primario esencial

**PROPOSITO:** Gestionar Cuentas

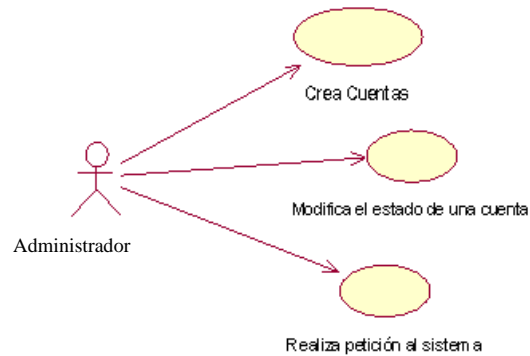
**VISION GENERAL:** Establecimiento de los niveles de usuario según su tipo

**REFERENCIAS:**

**CURSO TIPICO DE EVENTOS**

**Tabla N°IV 1 Tabla Típica de Eventos**

<b>Actores</b>	<b>Sistema</b>
1. El Administrador crea cuentas	2. Almacena la Información ingresada (nombre y contraseña) y realizar validación.
2. El Administrador modifica el estado de una cuenta	3. Actualiza información
5. El Administrador realiza petición del sistema	6. Confirmar datos



**FiguraN° IV 3 Grafico Caso de Uso Administrador**

**Caso de Uso de Alto nivel Limitada**

**CASO DE USO #2**

**NOMBRE:** Limitada

**Código:** CUAD

**ACTORES:** Limitada

**TIPO:** Primario

**PROPÓSITO:** Ingreso, actualización de la información, Generar reportes

**VISIÓN GENERAL:** Obtención de información y reportes

**REFERENCIAS:**

**CURSO TIPICO DE EVENTOS**

**Tabla N°IV 2 Curso típico de Eventos Limitada**

<b>Actores</b>	<b>Sistema</b>
1. El usuario limitado ingresa al sistema	2. Solicitar nombre, contraseña y realizar validación.
4. El usuario limitado accede al sistema visualiza información, ingresa, actualiza información y obtiene reportes	5. Espera una solicitud
5. El usuario limitado escoge un reporte	6. Genera reporte solicitado
7. El usuario limitado visualiza reporte generado	8. Espera confirmación



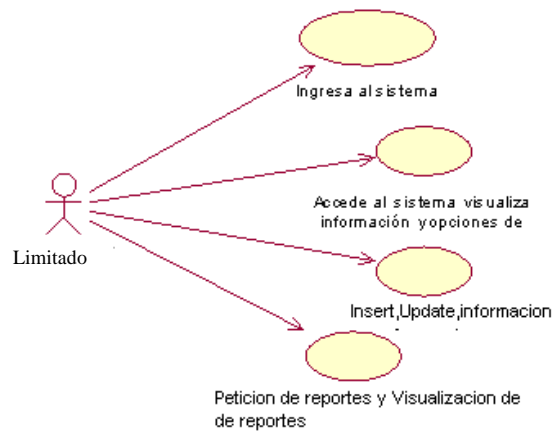


Figura N° IV 4 Grafico Caso de uso User Limitado

### Casos de uso de Alto nivel Usuario

#### CASO DE USO #3

**NOMBRE:** Usuarios

**Código:** CUS

**ACTORES:** Usuarios

**TIPO:** Primario

**PROPÓSITO:** Generar reportes

**VISIÓN GENERAL:** Obtención de información y reportes

**REFERENCIAS:**

**CURSO TIPICO DE EVENTOS**

Tabla N°IV 3 Curso Típico de Eventos usuario

Actores	Sistema
1. El usuario accede al sistema visualiza información, y obtiene reportes	1. Espera una solicitud
3. El usuario escoge un reporte	4. Genera reporte solicitado

4. El usuario visualiza reporte generado	5. Espera confirmación
--	------------------------

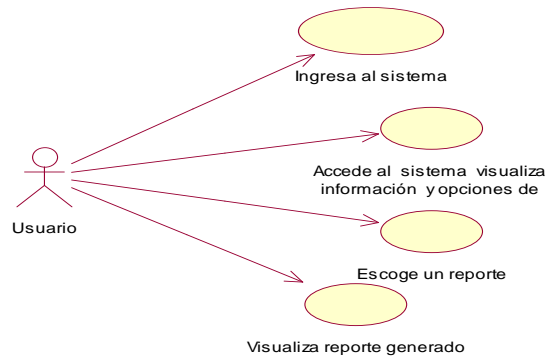


Figura N° IV 5 Grafico Caso de Uso Usuario

#### 4.4 Etapa 4. DISEÑO.

##### Definición de Informes e Interfaces de Usuario.

##### 4.4.1 Definición de la información de la interfaz de usuario.

La interfaz que se presenta en el sistema para la interacción con el usuario es en forma de página Web, con la introducción del concepto de formulario, por lo que para acceder al sistema es necesario contar con un browser o navegador.

##### 4.4.2 Lenguaje de comunicación.

###### a. Comunicación con el usuario

Al ser concebido como una aplicación Web el sistema SISALFA es de fácil acceso y manejo.

###### b. Nivel Arquitectónico

El entorno propuesto posee seguridad a nivel de arquitectura de aplicación como podemos observar en la figura a continuación.

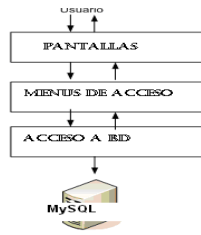
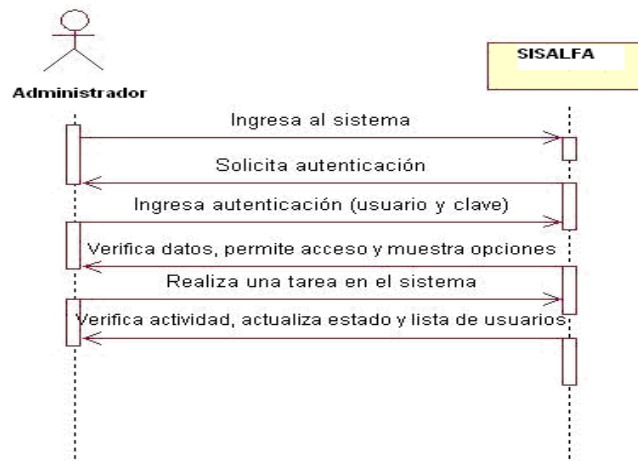


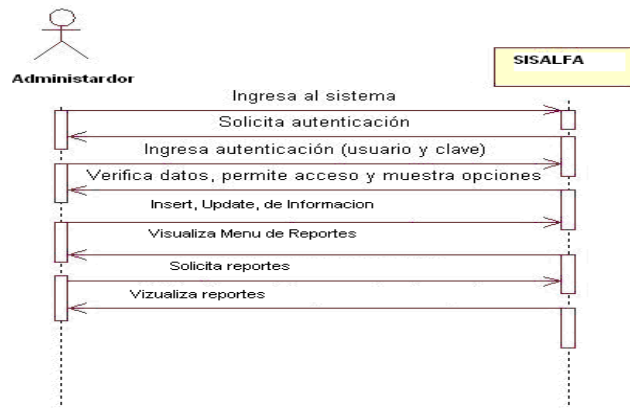
Figura N° IV 6 Nivel Arquitectónico

#### 4.4.3 Diagramas de Interacción

##### 4.4.3.1 Diagramas de Secuencia



FiguraN° IV 7 Diagrama de secuencia Verificación de usuarios logueados



FiguraN° IV 8 Diagrama de secuencia Administrador

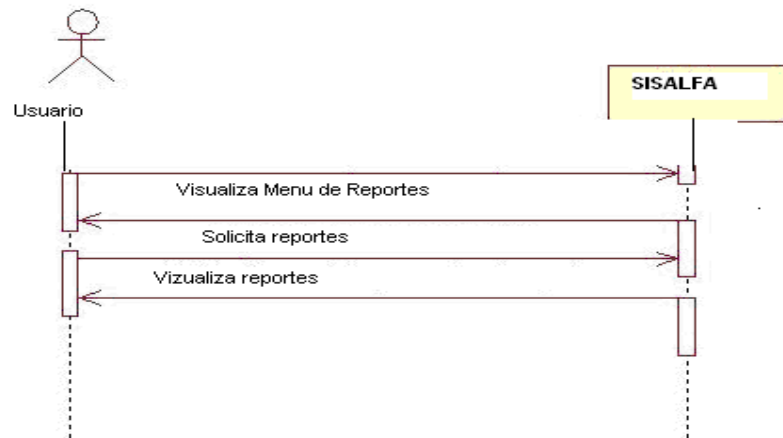


Figura N° IV 9 Diagrama de secuencia Usuario

#### 4.4.3.2 Diagramas de Colaboración

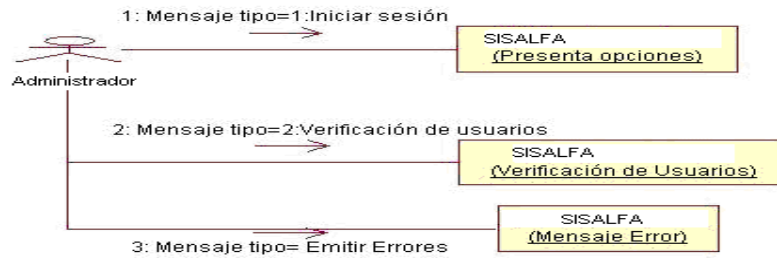


Figura N° IV 10 Diagrama de colaboración Verificación de usuarios logueados

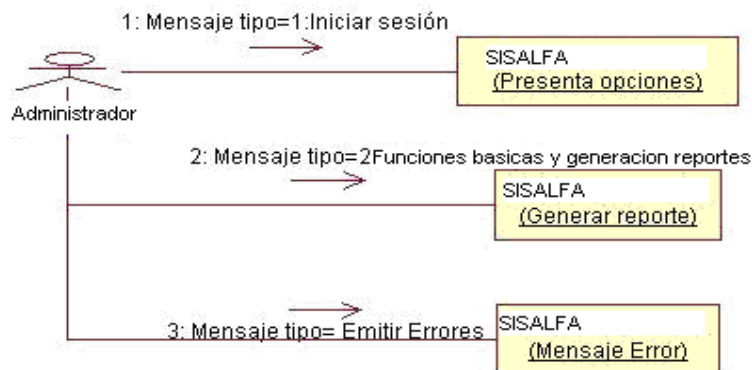


Figura N° IV 11 Diagrama de colaboración Administrador

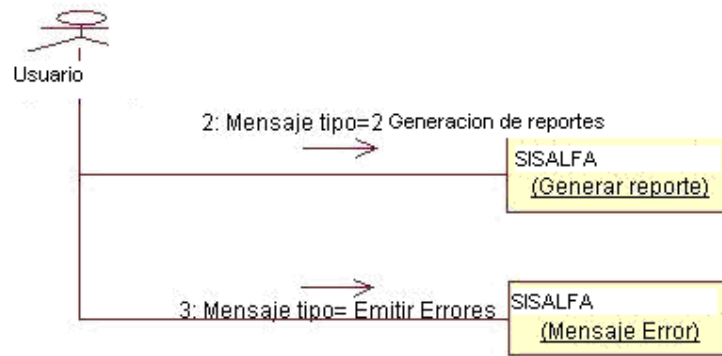


Figura N° IV 12 Diagrama de colaboración Verificación de servicios

#### 4.4.3.3 Diagramas de Calles

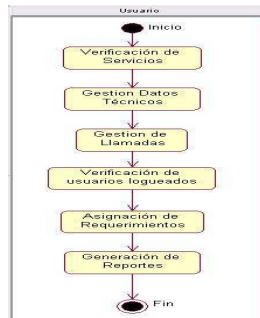


Figura N° IV 13 Diagrama de calles

#### 4.4.3.4 Diagramas de Base de Datos

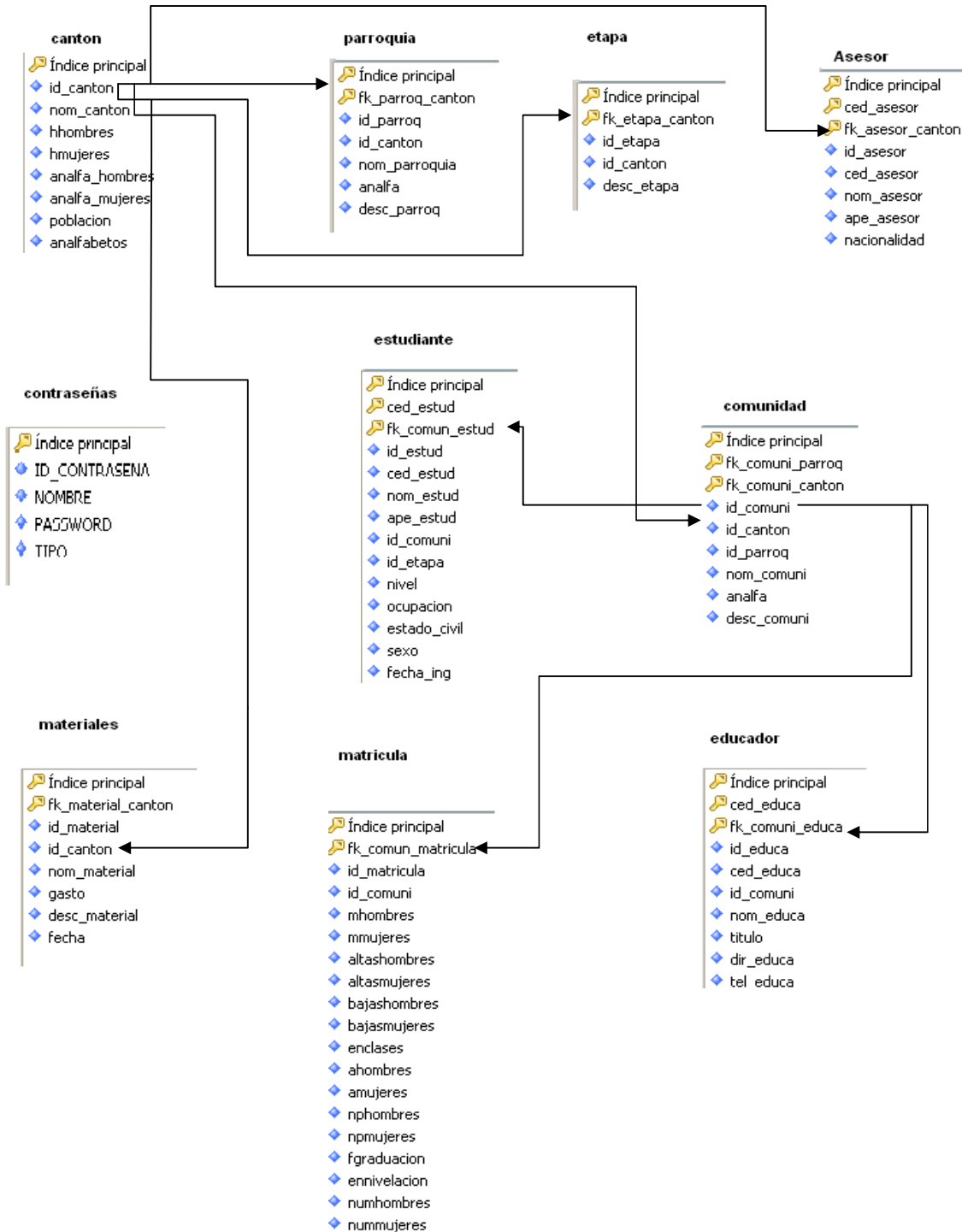


Figura N° IV 14 Diagrama de Base de Datos

#### 4.4.3.5 Diagramas de Despliegue

### Diagrama de Componentes

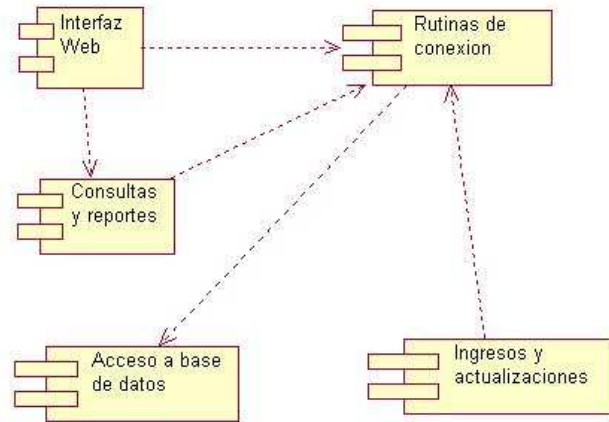


Figura N° IV 15 Diagrama de componentes

### Diagrama de Nodos

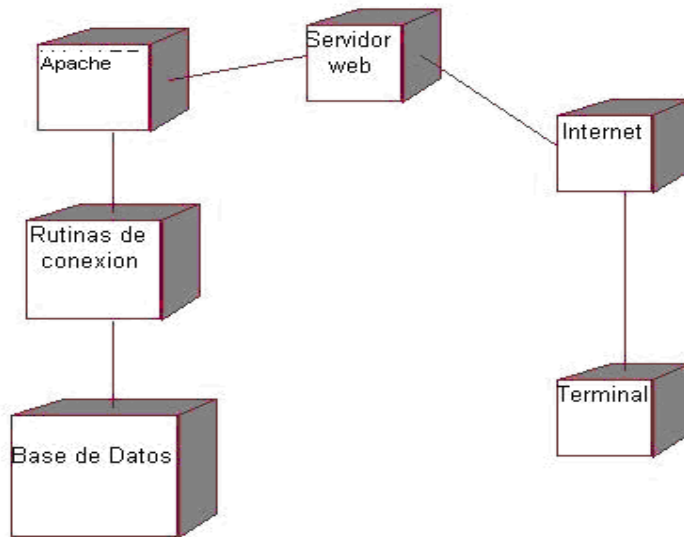


Figura N° IV 16 Diagrama de nodos

## **4.5 Etapa 5.- Implementación y Pruebas**

### **Definición de Estándares de Programación**

Para realizar la codificación del sistema se han definido los siguientes estándares:

El nombre de las tablas se encuentra con letras minúsculas.

El nombre de los campos empieza con una letra minúscula, seguido de una línea hacia abajo en el caso de más de una palabra.

Los métodos que interactúan con la base de datos reciben como parámetros de entrada:

- Connection: Representa la cadena de conexión hacia la base de datos.
- Lista de parámetros en el caso de ser necesaria.

Los métodos que interactúan con la base de datos en el caso de funciones de inserción, deben devolver como resultado el código del campo que ha sido insertado.

### **Pruebas**

Para asegurar el correcto funcionamiento del Sistema se han probado sus métodos de forma independiente, enviando datos de entrada desde el código, para luego obtener los a través de los diferentes métodos para realizar consultas.

Se han probado especialmente todas las funciones para validaciones de datos.

### **Prueba de Módulos y del Sistema**

Las pruebas finales consistieron en verificar que la información ingresada se vea inmediatamente reflejada en las consultas del sistema, esto sirve para comprobar que la información se está registrando correctamente en la base de datos.



Se provocaron errores intencionales para verificar el correcto funcionamiento del sistema, así como de las funciones de validación de datos, como por ejemplo:

- Realizar consultas a tablas vacías
- Ingresar campos incorrectos

Tratar de ingresar información diferente al tipo de dato correcto, como tratar de ingresar texto en los campos que son numéricos, ingresar formatos de fechas diferentes, tratar de ingresar datos técnicos en servicios que no admiten esta información.

## CONCLUSIONES

1. El Análisis Comparativo realizado en el presente trabajo permitió determinar las prestaciones que tiene el lenguaje de programación Web Perl con relación al lenguaje de programación Web Python.
2. Debemos destacar que tanto Perl como Python son lenguajes que se encuentran en constante evolución debido a que son Open Source y también a que las comunidades de programadores de cada lenguaje aportan con ideas para el mejoramiento del desarrollo de aplicaciones.
3. Tanto Perl como Python son lenguajes excelentes para desarrollar aplicaciones Web. Ambos lenguajes poseen facilidades y prestaciones para los desarrolladores.
4. Sus diferencias no son muy notables como es el caso del número de líneas de código ya que en varios casos la diferencia es de una línea. Esto ha llevado a la determinación de que son lenguajes primos hermanos. Calificativo utilizado por varios autores de artículos de los lenguajes.
5. Debemos anotar que por los resultados obtenidos la tecnología Perl es la más eficiente (en nuestro caso) para el desarrollo de la aplicación SYSALFA ya que Perl se muestra más sólido, y con un alto grado de madurez y confiabilidad, con relación a Python respectivamente.
6. Además debemos mencionar que Perl sirvió de inicio para la creación del lenguaje Python.

## RECOMENDACIONES

1. Como recomendación mencionaremos que debemos analizar la estructura y los sistemas informáticos existentes antes de decidir la utilización de una determinada tecnología, ya que esta debe siempre estar acorde con el entorno en el cual se va a desempeñar.
2. Revisar la información relacionada con el desarrollo de los módulos en cada una de los lenguajes analizados, para que pueda servir de guía en la construcción de aplicaciones Web.
3. También debemos tomar en cuenta los requerimientos de los usuarios finales de nuestra aplicación, ya que esto nos evitara presentar sistemas con falencias y de disgusto para los usuarios terminares.
4. Así mismo si existe experiencia en la problemática a ser desarrollada se debe sugerir cambios en la forma de recolección de datos. Esto facilitara la subida de datos a la base teniendo como antecesora una plantilla estándar y de múltiple uso.
5. Para la realización de sistemas Web con las características similares o iguales a SYSALFA recomendamos la utilización de Perl, ya que este lenguaje aprovecha de mejor manera la utilización de recursos, para este caso. Aunque no debemos descartar la utilización de Python para otras aplicaciones puesto que esta tecnología (Python) también presenta prestaciones para el desarrollo de aplicaciones. La decisión debe depender del problema a resolver, tomando en cuenta las diversas venjatas y desventajas que presentas ambas tecnologías.

## RESUMEN

La presente investigación esta orientada al estudio comparativo entre las Tecnologías de Código Libre Perl y Python para el Desarrollo de aplicaciones Web, aplicado al Programa de Alfabetización, que lleva a efecto el Honorable Consejo Provincial de Chimborazo.

De acuerdo al estudio comparativo y mediante los parámetros de comparación Acceso a Base de Datos, Líneas de Código, Portabilidad, Interfaz de Usuario, se determinó un **93.75%** para la Tecnología Perl y **90.62%** para la Tecnología Python respectivamente; y por ende se concluyó que la Tecnología Perl es la mejor opción ya que es multiplataforma y se puede utilizar para el desarrollo de aplicaciones Web con código libre.

Para la implementación y pruebas de módulos se utilizó Perl 5.8.5, Python 2.3.4, MySQL 4.1.2, Sistema Operativo Centos Server 4.3, XP SP2, donde cada uno de estos software con sus distintas versiones fueron utilizados tanto en Windows como en Linux para su respectivo análisis de comparación.

Se recomienda utilizar este Sistema para el Control del Programa de Alfabetización, ya que gracias a este sistema se logrará reducir tiempos y errores, que permitirá obtener la información actualizada y ordenada.

## SUMMARY

The current investigation is aimed at comparative study between Technologies Code Free Perl and Python for the Development of Web applications, applied to the Literacy Programme, which takes effect on the Honourable Provincial Council of Chimborazo.

According to the study and through the comparative benchmarks Access Database, lines of code, portability, User Interface, was determined by 93.75% to 90.62% Perl and Technology for Technology Python respectively, and therefore it was concluded Technology that Perl is the best choice since it is multiplatform and can be used for developing Web applications with open source.

For the implementation and tests of modules used Perl 5.8.5, Python 2.3.4, 4.1.2 MySQL, Centos Server 4.3 operating system, XP SP2, where each of these software with its various versions were used both on PC and Linux for their respective comparative analysis.

It is recommended to use this system for the Control of the Literacy Programme, as a result of this system will be achieved reduce times and errors, which will provide updated information and orderly.

## GLOSARIO

**Applet:** Es una aplicación especial que se ejecuta dentro de un navegador o browser.

**Browser:** Navegador de Internet.

**Copyleft:** Copyleft es un concepto general, para proteger actualmente un programa. No permiten a los redistribuidores agregar ninguna restricción adicional cuando éstos redistribuyen o modifican el software.

**Copyright:** Software protegido por leyes de un país, software que posee dueño

**HTTP:** (Hiper Text Transfer Protocol). Protocolo de transferencia de Hipertexto. Es un protocolo de aplicación con la sencillez y velocidad necesaria para sistemas de información distribuidos, colaborativos y de diferentes medios.

**Intranet:** Una intranet es básicamente una Internet diseñada para ser utilizada internamente en una compañía. La principal diferencia entre Internet e Intranet es que la primera es pública y la segunda es privada.

**Linux:** Sistema operativo Open Source, de utilización libre y sin restricciones en su manejo, se lo denomina software libre.

**MySQL:** Es un sistema gestor de bases de datos, se trata de una de las bases de datos más rápida actualmente.

**PHP:** Lenguaje de Programación de uso libre, es decir un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor.

**Perl:** Lenguaje de Programación de uso libre para el desarrollo de aplicaciones Web, por ende es un lenguaje de propósito general

**Python:** Lenguaje de programación orientado a objetos de uso libre para el desarrollo de aplicaciones Web.

**Windows:** Sistema Operativo donde Microsoft es el más grande informático que produce y comercializa Windows, el sistema operativo que usa el 90% de los ordenadores personales de todo el mundo.

## BIBLIOGRAFÍA

### LIBROS

- **STEPHEN**, Like. Mastering Perl/Tk. 4ta ed. Texas: McGraw-Hill, 2005 pp. 345
- **MARK**, Lutz. Programming Python 2da ed. Texas: Illustrater, 2006 pp.457
- **MERELO GUERVOS**, Juan . Perl en Castellano. 1ra. ed. Texas: Web DeveloPpyent Perl, 2007. pp. 203.
- **MAXWELL**, S. Centos Server: Herramientas para la administración de redes. 3ra. ed. Madrid: McGraw-Hill, 2004. pp. 497.
- **PETERSON**, R. Red Hat Linux, Manual del Administrador. 3ra. ed. Madrid: McGraw-Hill, 2004. pp. 396.
- **PRESSMAN**, R. Ingeniería de Software. 4ta. ed. Madrid: McGraw-Hill, 2000. pp. 800.



## **BIBLIOGRAFÍA INTERNET**

- **LA ENCICLOPEDIA LIBRE PERL**

<http://es.wikipedia.org/wiki/Perl>

(2007/07/23)

- **TUTORIAL DE PERL EN ESPANOL**

<http://geneura.ugr.es/~jmerelo/tutoperl/>

(2005/02/13)

- **PROGRAMACION EN PERL**

[http://es.wikibooks.org/wiki/Programaci%C3%B3n\\_en\\_Perl](http://es.wikibooks.org/wiki/Programaci%C3%B3n_en_Perl)

(2006/11/17)

- **INTRODUCCION A LA PROGRAMACION PYTHON**

<http://marmota.act.uji.es/MTP/pdf/python.pdf>

(2007/05/28)

- **PROGRAMACION PYTHON**

<http://www.javielinux.com/articulos/pygimp/c120.htm>

(2007/12/05)

- **INTRODUCCION A PERL**

<http://www.ulpgc.es/otros/tutoriales/perl/cap1.htm>

(2007/01/03)

- **PYTHON VS. OTROS LENGUAJES**

<http://users.servicios.retecal.es/tjavier/python/intpython/x164.html>

(2007/09/14)

**ANEXOS**

## **ANEXO A**

### **SRS**

#### **Especificación de Requerimientos de Software (SRS)**

##### **1. Introducción**

###### **1.1 Objetivos del documento**

- Realizar una especificación detallada de los requerimientos con los que debe cumplir el desarrollo del Sistema SISALFA, especificación que servirá tanto al cliente como al desarrollador como referencia para la implementación del sistema.
- Definir explícitamente entidades, entradas, salidas y procesos a ser realizados en la implementación del sistema para proporcionar una visión amplia y detallada de su funcionalidad.
- Efectuar una planificación detallada de las actividades correspondientes a cada etapa del proceso de desarrollo de software elegido por parte del desarrollador.

###### **1.2 Audiencia a la que va dirigido**

El presente documento está dirigido para la siguiente audiencia:

- Desarrollador del sistema
- Ejecutivos del departamento de Gestión Social del H. Consejo Provincial.
- Director del departamento de Gestión Social.
- Coordinadores, educadores, asesores del programa de alfabetización.

### 1.3 Alcance

El Sistema de SISALFA tiene como misión realizará las siguientes tareas:

- Registrar la información personal de los alumnos del programa.
- Registrar la información de los asesores, educadores, coordinadores del programa.
- Registrar el presupuesto invertido en cada programa.
- Consultar los datos de los estudiantes
- Actualizar la información existente.

El sistema realizará única y exclusivamente aquello que está considerado dentro de los puntos mencionados anteriormente.

Los principales beneficios que el Sistema SISALFA proporcionará al consejo son:

- Disponer de una aplicación Web desarrollada en código libre.
- Tener reportes de la marcha del programa de alfabetización para la toma de decisiones..
- Administrar adecuadamente toda la información relacionada con los estudiantes a los cuales va dirigido el programa de alfabetización de forma segura.

### 1.4 Definiciones, acrónimos y abreviaturas

**SISALFA:** Sistema de Alfabetización

**SRS:** Especificación de Requerimientos de Software.

**Estudiante:** Persona jurídica que recibe el programa (persona alfabetizada).

## **1.5 Referencias**

- IEEE Std 830 – Software Engineering Standards Committee of the IEEE Computer Society.
- IEEE Std 610.12 – 1990, IEEE Standard Glossary of Software Engineering Terminology (ANSI).
- IEEE P1233, October 1993, Draft Guide to Developing Systems Requirements Specifications.
- IEEE P1233/D3 December 1995, Recommended Practice for Software Requirements.

## **2. Descripción General**

### **2.1 Perspectiva del producto**

SISALFA es un producto que realiza ingresos, consultas y actualizaciones de la base de datos de los estudiantes del programa de alfabetización, y luego las visualiza en la aplicación Web, además permite la generación de reportes estadísticos de estudiantes en cada cantón, parroquia o comunidad.

#### **2.1.1 Interfaces de Sistema**

1. El sistema debe interactuar con el usuario y visualizar la información en forma de páginas Web mediante un browser.
2. Almacenar los datos en un servidor de base de datos MySQL4.1, sobre un equipo Pentium IV con Linux Centos 4.3.
3. Consultar la información de estudiantes y presupuesto del servidor de base de datos de la institución.

4. El sitio Web debe estar alojado sobre un servidor Apache Tomcat instalado sobre Linux CentOS versión 4.3.

### **2.1.2 Limitaciones de Memoria**

No se establecen limitaciones en cuanto a la cantidad de la memoria, tanto primaria como secundaria que el sistema deba utilizar.

### **2.1.3 Hardware a usar**

- Servidor Linux Centos 4.3 (Servidor de Base de Datos).
- Servidor Web.

## **2.2 Funciones del producto**

Las funciones que el sistema SISALFA realizará son:

- Gestión de la información de los estudiantes del programa de alfabetización
- Gestión de estudiantes matriculados en el programa.
- Validación de datos
- Consulta de información técnica de los estudiantes.

La gestión de la información técnica de los estudiantes involucra las siguientes tareas:

- Ingreso
- Actualización y
- Consulta de información

## **2.3 Características de los usuarios**

Las personas que utilizarán el sistema para uno u otro propósito poseen las siguientes características:

### **2.3.1 Administrador y Limitada**

Personas encargadas de realizar el control y administración de la aplicación dentro de la institución.

#### **2.3.1.1 Nivel educacional.**

Los Administradores y Limitadas del sistema cuentan con un nivel de Educación Superior lo que garantiza que el sistema conserve una administración, eficiente, confiable y eficaz.

#### **2.3.1.2 Experiencia.**

La experiencia se considera aceptable para poder desarrollar e implementar este sistema.

#### **2.3.1.3 Conocimientos Técnicos.**

Los Administradores y Limitadas del sistema cuentan con los conocimientos técnicos apropiados para mantener un buen nivel de operabilidad.

#### **Usuarios:**

Persona encargada de la utilización del sistema dentro de la institución.

### **2.3.2.1 Nivel educativo.**

El nivel de educación de los usuarios del sistema se limita a personas que posean los conocimientos apropiados para el manejo de un equipo de cómputo, los conocimientos para el manejo del nuevo sistema serán impartidos por el desarrollador del sistema.

### **2.3.2.2 Experiencia.**

Los usuarios del sistema cuentan con una experiencia intermedia en el manejo de sistemas interactivos (como Windows).

### **2.3.2.3 Conocimientos Técnicos.**

Los usuarios deberán poseer conocimientos básicos en el manejo del sistema.

## **2.4 Limitaciones Generales**

- Los datos deben ser manejados por un servidor Linux instalado sobre un equipo Linux Centos 4.3.
- El sitio Web debe alojarse sobre un servidor Apache Tomcat instalado sobre un servidor Linux Centos versión 4.4.
- El diseño del sitio debe sujetarse a la utilización de colores y logotipos de la institución.

## **2.5 Suposiciones y dependencias**

- El browser del cliente admitirá la ejecución de scripts de cliente.
- El usuario debe tener conocimientos acerca de la navegación en Internet.



## **2.6 Requisitos para futuras versiones**

No se han establecido requisitos para futuras versiones del software.

## **3. Requisitos Específicos**

### **3.1 Requisitos de interfaz externos**

#### **3.1.1 Interfaces de usuario**

- El sistema deberá presentar una interfaz Web de fácil navegación.
- El sistema debe informar de errores y/o omisiones en el ingreso de datos por parte del usuario.
- Permitir la corrección de información mal ingresada.
- El texto de los componentes de interfaz debe ser claro y comprensible

##### **3.1.1.1 Diseño de Entradas**

El ingreso de datos debe cumplir con los siguientes requisitos:

- La información debe ser clara y concreta.
- Deben existir listas de selección.
- Se deben evitar al máximo errores de ingreso.
- Debe permitir valores por defecto.
- Verificar la validez de los datos ingresados.
- Informar de los errores en el ingreso de datos.

##### **3.1.1.2 Salidas del Sistema**

El sistema SISALFA proporcionará salidas mediante páginas Web a cerca de información referente a:

- Información de los estudiantes.
- Información de los estudiantes matriculados ordinaria y extraordinariamente.
- Información del presupuesto invertido en la ejecución de cada programa.

### **3.1.1.3 Diseño de salidas**

El diseño de salidas que deberá presentar el sistema deberá cumplir con el formato que se muestra en el **Anexo**.

### **3.1.2 Interfaces hardware**

No existen interfaces especiales de hardware

### **3.1.3 Interfaces software**

SISALFA debe interactuar con las siguientes interfaces software:

- Servidor Web.
- Servidor de base de datos MySql
- Protocolo TCP/IP
- Sistemas operativos Linux

Las paginas Web accesibles desde la Internet deben estar alojadas en un servidor Apache Tomcat sobre Linux CentOS versión 4.3, mientras que las páginas que se

visualizan internamente a través de la Intranet pueden ser provistas por el equipo mencionado.

#### **3.1.4 Interfaces de comunicaciones**

Ya que SISALFA trabaja en un ambiente Web, el protocolo de comunicaciones que se va a utilizar es HTTP que forma parte del protocolo TCP/IP.

### **3.2 Requisitos funcionales**

Los requisitos con los que debe cumplir el sistema, se listan a continuación:

#### **Req01:** Agregar datos de un estudiante

**Objetivo:** Incluir en la base de datos la información de los datos de un estudiante.

**Entradas:** cedula, nombre, apellido...etc.

**Proceso:** El usuario ingresa los datos requeridos y pulsa Ingresar, a esto el sistema responde validando los datos y en caso de ser correctos son almacenados, si existen errores se muestran mensajes describiendo los mismos.

**Salida/Resultado:** Almacenamiento de los datos de un estudiante.

**Mensajes:** Debe Ingresar un valor válido para: (campos que no tengan valores o que tengan valores incorrectos)

#### **Req02:** Actualizar datos técnicos de un estudiante.

**Objetivo:** Modificar unos o más registros de datos de un determinado estudiante.

**Entradas:** Datos que permitan ser modificados.

**Proceso:** El usuario selecciona un estudiante para modificar sus datos, y el sistema presenta la información respectiva para ese estudiante, en los campos que pueden ser modificados se cambia la información y se pulsa Modificar, a esto el sistema responde validando los datos y de ser correctos son almacenados, caso contrario se muestran mensajes de error.

**Salida/Resultado:** Actualización de los datos del estudiante.

**Mensajes:** Debe Ingresar un valor válido para: (campos que no tengan valores o que tengan valores incorrectos)

**Req03:** Consultar datos de un estudiante

**Objetivo:** Consultar los datos de un estudiante.

**Entradas:** Cedula.

**Proceso:** El usuario la cedula del estudiante a consultar los datos y el sistema muestra la información de los datos del estudiante seleccionado.

**Salida/Resultado:** Datos del estudiante.

**Mensajes:** Este servicio no admite otros datos.

**Req04:** Verificar los usuarios logueados.

**Objetivo:** Verificar los usuarios que se encuentran logueados en el sistema.

**Entradas:** Usuario y clave.

**Proceso:** El usuario ingresa al sistema, ingresa usuario y clave y el sistema verifica la actividad, permite el acceso, muestra opciones y actualiza el estado a logueado.

**Salida/Resultado:** Opciones del sistema.

**Req05:** Generación de reportes.

**Objetivo:** Generar reportes estadísticos y porcentuales de estudiantes.

**Entradas:** Ninguna

**Proceso:** El usuario ingresa al sistema y selecciona las opciones de reportes existentes, a esto el sistema responde mostrando el reporte seleccionado.

**Salida/Resultado:** Parámetros para reportes.

**Req6:** Consultar los datos de un Estudiante.

**Objetivo:** Consultar la información de un determinado estudiante.

**Entradas:** Cedula.

**Proceso:** El usuario busca un estudiante y en la parte derecha del estudiante pulsa el link que aparece, el sistema responde mostrando la información del estudiante seleccionado.

**Salida/Resultado:** Datos del estudiante.

### **3.3 Restricciones de diseño**

- El diseño de la arquitectura del sistema debe asegurar el manejo de los datos, la lógica de la aplicación y el manejo de su interfaz se realicen de forma separada e independiente entre ellas.
- El manejo de los datos lo debe realizar un servidor de base de datos Mysql 4.1, la lógica de la aplicación y la interfaz deben residir en un Servidor Web.

### **3.4 Atributos del sistema software**

El sistema SISALFA cuenta con los siguientes atributos:

### **3.4.1 Seguridad**

Las operaciones de Ingreso, Actualización de la información que presenta el sistema pueden ser accesibles desde la Intranet de la Institución y a través del Internet, únicamente podrán ser utilizadas por personal de encargado del programa, el mismo que cuente con los permisos necesarios para su utilización.

### **3.4.2 Disponibilidad**

La disponibilidad representa la potencialidad y la capacidad que tiene SISALFA para ser utilizado, comprendido y operado por los usuarios, además lo atractivo que debe ser, incluye características como comunicatividad, operabilidad, facilidad de aprendizaje, todas estas hacen que el sistema sea agradable de usar.

### **3.5 Otros Requisitos**

- **Base de Datos**

El sistema debe almacenar todos los datos que maneja en una base de datos Mysql4.1.

- **Servidores**

Se debe utilizar un servidor Web.

- **Operaciones:**

Las principales operaciones que el sistema va a realizar con la información son las siguientes:

- Inserción.
- Actualización.
- Consultas.

#### 4. Programación Temporal

Nombre de tarea	Duración	Comienzo	Fin
<b>SISALFA</b>	45 días	lun 02/10/07	vie 01/12/07
Planteamiento del Problema	2 días	lun 02/10/07	mar 03/10/07
Peticiones de Usuario	2 días	mié 04/10/07	jue 05/10/07
Reunión con el personal	1 día	vie 06/10/07	vie 06/10/07
Entrevista con el personal	1 día	lun 09/10/07	lun 09/10/07
Toma de Decisiones	2 días	mar 10/10/07	mié 11/10/07
Determinación de Riesgos Técnicos	1 día	jue 12/10/07	jue 12/10/07
Especificación de Recursos Económicos	1 día	vie 13/10/07	vie 13/10/07
Especificación de Recurso Humano	1 día	lun 16/10/07	lun 16/10/07
Especificación de Recursos Hardware	1 día	mar 17/10/07	mar 17/10/07
Especificación de Recursos Software	1 día	mié 18/10/07	mié 18/10/07
Especificación de Recursos Físicos	1 día	jue 19/10/07	jue 19/10/07
Verificación de Recursos	1 día	vie 20/10/07	vie 20/10/07
Estrategias de Solución	1 día	lun 23/10/07	lun 23/10/07
Selección del Motor de Base de Datos	1 día	mar 24/10/07	mar 24/10/07
Selección de Herramientas de Implementación	1 día	mié 25/10/07	mié 25/10/07
Selección de Herramientas para Interfáz	1 día	jue 26/10/07	jue 26/10/07
Selección de Herramientas de Apoyo	1 día	vie 27/10/07	lun 27/11/07
Diseño Conceptual	3 días	lun 30/10/07	mié 01/11/07
Diseño Físico	2 días	jue 02/11/07	vie 03/11/07
Implementación del Sistema	15 días	lun 06/11/07	vie 24/11/07
Seguridad en la Aplicación	1 día	lun 27/11/07	lun 27/11/07
Integración	2 días	mar 28/11/07	mié 29/11/07
Presentación del Prototipo	1 día	jue 30/11/07	jue 30/11/07
Aprobación Gerencial	1 día	vie 01/12/07	vie 01/12/07
Pruebas, Validaciones y Control	3 días	lun 04/12/07	mié 06/12/07
Presentación Final del Proyecto	5 días	jue 07/12/07	mié 13/12/07
Capacitación del Personal	5 días	jue 14/12/07	mié 20/12/07
Entrega del sistema y Derechos del mismo	2 días	jue 21/12/07	vie 22/12/07
Mantenimiento del Sistema	1 día	mar 26/12/07	mar 26/12/07

## ANEXO B

### CODIGO PERL

- **ACCESO A BASE DE DATOS**

```
use DBI;
my $db = DBI->connect('DBI:mysql:consejo:localhost','root','root')|| die
"ERROR CONECCION : $DBI::errstr";
$db->disconnect();
```

- **INSERCIÓN DE REGISTROS**

```
my $sth1 = $db->prepare("INSERT INTO
canton(nom_canton,hombres,hmujeres,poblacion,analfa_hombres,analfa_mujer
es,analfabetos,desc_canton) VALUES (?, ?, ?, ?, ?, ?, ?)");
$sth1->execute($form{'nom_canton'},$form{'hombres'},$form{'hmujeres'},$for
m{'hombres'}+$form{'hmujeres'},$form{'analfa_hombres'},$form{'analf_muje
res'},$form{'analfa_hombres'}+$form{'analfa_mujeres'},$form{'desc_canton'});
```

- **CONSULTA Y ACTUALIZACIÓN DE REGISTROS**

```
use DBI;
my $sth1 = $db->prepare("UPDATE canton SET
nom_canton=?,hombres=?,hmujeres=?,poblacion
=?,analfa_hombres=?,analfa_mujeres=?,analfabetos=?,desc_canton=? WHERE
id_canton=?");
$sth1->execute($form{'nom_canton'},$form{'hombres'},$form{'hmujeres'},$f
orm{'hombres'}+$form{'hmujeres'},$form{'analfa_hombres'},$form{'analfa_m
ujeres'},$form{'analfa_hombres'}+$form{'analfa_mujeres'},$form{'desc_canton'
},$CLIENTE3);
$sth1->finish;
```

- **ELIMINACIÓN DE REGISTROS**

```
use DBI;
my $sth1 = $db->prepare("DELETE FROM canton WHERE id_canton= ?");
$sth1->execute($c);
$sth1->finish;
```

- **VIZUALIZACIÓN DE CANTONES**

```
$$sqlStatement = "SELECT * FROM canton ORDER BY canton.nom_canton";
```



```

if ($db->Sql($SqlStatement)) {
    print "la sentencia SQL ha fallado.\n";
    print "Error: " . $db->Error() . "\n";
    exit;
}
while($db->FetchRow())
{
    %Data = $db->DataHash();
    print "<TR>";
    print "<font face=Arial size=1>";
    print "<TD align=center><B>".$Data{"nom_canton"}."</B><TD
align=center>".$Data{"hhombres"}."<TD
align=center>".$Data{"hmujeres"}."<TD
align=center>".$Data{"poblacion"}."<TD
align=center>".$Data{"analfa_hombres"}."<TD
align=center>".$Data{"analfa_mujeres"}."<TD
align=center>".$Data{"analfabetos"}."<TD
align=center>".$Data{"desc_canton"}."<Br>\n";
    print "</font>";
}

```

## CODIGO PYTHON

- **ACCESO A BASE DE DATOS**

```

<%
import MySQLdb, time, string, _mysql, _mysql_exceptions
#funcion que permite la conexion a la base de datos y la ejecucion de comandos
sql

def executeSQLCommand(cursor, command):
    rowSet = []
    command = string.strip(command)
    if len(command):
        try:
            cursor.execute(command) # Ejecuta el comando
            if string.lower(command).startswith('select'): # si es select
                lines = cursor.fetchall() # recuperar todos los resultados
                for line in lines:
                    row = []
                    for column in line:
                        row.append(column)
                    rowSet.append(row)
        except _mysql_exceptions.ProgrammingError, e:
            print e
    return rowSet

```

```

%>
<%
db = MySQLdb.connect(host="localhost",user="root",passwd="root",
db="consejo")
cursor = db.cursor()
del cursor
%>

```

- **CODIGO PARA INSERTAR UN NUEVO REGISTRO**

```

<%
db = MySQLdb.connect(host="localhost",user="root",passwd="root",
db="consejo")
cursor = db.cursor()
executeSQLCommand(cursor,"insert into
canton(nom_canton,hombres,hmujeres,poblacion,analfa_hombres,analfa_mujer
es,analfabetos,desc_canton) values('%s','%s','%s','%s','%s','%s','%s','%s')" %
(form['nom'].capitalize(),form['hhombres'].capitalize(),form['hmujeres'].capitaliz
e(),form['habita'].capitalize(),form['analfa_hombres'].capitalize(),form['analfa_m
ujeres'].capitalize(),form['analfa'].capitalize(),form['desc'].capitalize()))
%>

```

- **ACTUALIZACION Y CONSULTAS DE REGISTROS**

```

<%
db = MySQLdb.connect(host="localhost",user="root",passwd="root",
db="consejo")
cursor = db.cursor()
executeSQLCommand(cursor,"update canton set hhombres= '%s',hmujeres=
'%s',poblacion=
'%s',analfa_hombres='%s',analfa_mujeres='%s',analfabetos='%s',desc_canton='
%s' where nom_canton ='%s'" %
(form['hhombres'],form['hhombres'],form['habita'],form['analfa_hombres'],form['
analfa_mujeres'],form['analfa'],form['desc'],form['nom']))
%>

```

- **ELIMINACION DE REGISTROS**

```

<%
db = MySQLdb.connect(host="localhost",user="root",passwd="root",
db="consejo")
cursor = db.cursor()
executeSQLCommand(cursor,"delete from canton where nom_canton='%s'" %
form['nom'])
%>

```

- **VIZUALIZACION DE CANTONES**

```
<%
import MySQLdb, time, string, _mysql, _mysql_exceptions
#funcion que permite la conexion a la base de datos y la ejecucion de comandos
sql
def executeSQLCommand(cursor, command):
    rowSet = []
    command = string.strip(command)
    if len(command):
        try:
            cursor.execute(command) # Ejecuta el comando
            if string.lower(command).startswith('select'): # si es select
                lines = cursor.fetchall() # recuperar todos los resultados
                for line in lines:
                    row = []
                    for column in line:
                        row.append(column)
                    rowSet.append(row)
        except _mysql_exceptions.ProgrammingError, e:
            print e
    return rowSet

%>
<%-- IMPORTANTE estamos definiendo una funcion que permita conectarnos
a una base de datos en MySQL--%>

<body>
<center><h1>Tabla de Datos</h1></center>
<center>
<table width="200" border="1">
<%
db = MySQLdb.connect(host="localhost",user="root",passwd="root",
db="consejo")
cursor = db.cursor()
#executeSQLCommand(cursor,"insert into usuario(nombreUsuario,password)
values('15','15')")
rowSet = executeSQLCommand(cursor, "select * from canton")
for row in rowSet:
    req.write("<tr>")
    req.write("<td> %s </td>" % (row[0]))
    req.write("<td> %s </td>" % (row[1]))
    req.write("<td> %s </td>" % (row[2]))
    req.write("<td> %s </td>" % (row[3]))
    req.write("</tr>")
del cursor
```

```
%>
</table>
</center>
</body>
```

## ANEXO C

### SENTENCIAS PYTHON Y PERL

#### SENTENCIAS EN PYTHON

```
<%
if x == y:
    # begin
%>
... some html ...
<%
# end
%>
```

- **Hola Mundo en Python**

```
<html>
<%
if form.has_key('name'):
    greet = 'Hola, %s!' % form['name'].capitalize()
else:
    greet = 'Hola Mundo!'
# end
%>
<h1><%= greet %></h1>
</html>
```

- **Estructuras de Control**

```
if x < 5 or (x > 10 and x < 20):
    print "El valor es correcto."
if x < 5 or 10 < x < 20:
    print "El valor es correcto."
for i in [1,2,3,4,5]:
    print "Pasada nº ", i
x = 10
while x >= 0:
    print "x todavía no es negativo."
    x = x-1
```

- **Sentencias en Perl**

```
print " Pensando un número...\n";
$numero = rand(100);
print> " Introduce un numero del 1 al 100:";
$entrada = <STDIN>;
if($entrada == $numero) {print "Has acertado el numero\n";}
if($entrada > $numero) {print "El numero introducido es mayor que el
pensado.\n";}
if($entrada < $numero) {print "El numero introducido es menor que el
pensado.\n";}
print" El número pensado era:",$numero;
```

- **Estructuras de Control**

**if (expresión) bloque else bloque.**

Un ejemplo podria ser el siguiente:

```
print" Pensando un número...\n";
$numero = rand(100); print" Introduce un numero del 1 al 100:";
$entrada = <STDIN>;
if($entrada == $numero){
print"Has acertado el numero\n";}
else
{if($entrada > $numero){
print"El numero introducido es mayor que el pensado.\n";}
else
{print "El numero introducido es menor que el pensado.\n";}
}
```

**ANEXO D**

# **MANUAL DE USUARIO**

## **SISALFA - HCPCH**



**SISTEMA PARA EL CONTROL DEL PROGRAMA DE  
ALFABETIZACION “YO SI PUEDO” DEL HONORABLE CONSEJO  
PROVINCIAL DE CHIMBORAZO**

## INTRODUCCIÓN

Este manual de usuario le ayudará paso a paso a entender el funcionamiento del SISALFA, además de familiarizarse con el producto para que no tenga contratiempos en el uso del mismo.

Durante el desarrollo de esta guía encontrará la información necesaria para que pueda trabajar con seguridad y pueda obtener el máximo rendimiento del producto. Además mediante ilustraciones gráficas usted podrá comprender de mejor manera todos los procedimientos que debe realizar para no causar daños al sistema y al producto.

## REQUISITOS MÍNIMOS PARA LA INSTALACIÓN

Los requerimientos mínimos son importantes ya que de ellos depende la buena funcionalidad del sistema. Para lo cual en este capítulo se hará énfasis desde el tipo plataforma, librerías que deben estar instaladas previamente, conexión a la base de datos, etc. Tanto en la parte del Servidor como en el tipo de computador cliente.

## REQUERIMIENTO DE INFRAESTRUCTURA

Con respecto a la infraestructura y con referencia al Hardware es necesario cumplir con los siguientes requerimientos mínimos:

<b>DISPOSITIVO CLIENTE</b>	<b>REQUERIMIENTO MÍNIMO</b>
Procesador	PII o superior
Memoria	256 MB o superior
CD ROM	40X o superior
Espacio en Disco	20GB o superior
Tarjeta de Red	10/100
Puertos	USB 2.0
<b>DISPOSITIVO SERVIDOR</b>	<b>REQUERIMIENTO MÍNIMO</b>
Procesador	PIII o superior
Memoria	256 MB o superior

CD ROM	40X o superior
Espacio en Disco	20GB o superior
Tarjeta de Red	10/100
Puertos	USB 2.0

## REQUERIMIENTO DE PLATAFORMA

SEPROCH V1.0 es un sistema que fue desarrollado en base a la plataforma Linux

Centos Server 4.3. Por lo que se considera necesario el tener instalado como mínimo:

CLIENTE
Windows 98, XP, Vista, Linux
Internet Explorer 5 o superior

SERVIDOR		
Centos Server 4.3	PhPpyAdmin 2.11	Apache 2.x.x.
Perl 5.8.5	MySql 4.1.2	Python 2.3.4

## OPERACIONES DEL SISTEMA SISALFA

En este capítulo se detalla la forma de operación del sistema SISALFA en su versión y la forma de administración en la parte de usuarios, la misma que es muy frecuente y bien básica.

## INICIO DEL SISTEMA

Para poder ver el sistema en el explorador debemos escribir la dirección <http://www.sisalfa.gov.ec>, en el cliente que se encuentre en la intranet, caso contrario si no está el cliente en la intranet el sistema no se podrá visualizar. La aplicación en primera instancia nos mostrará la siguiente pantalla.





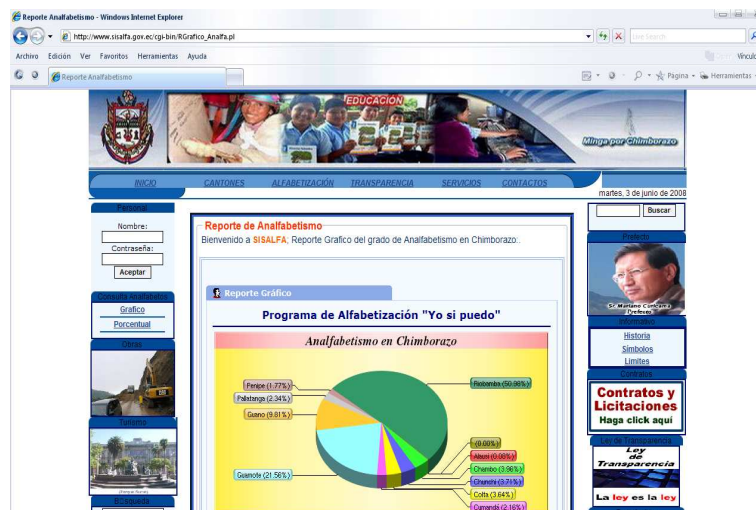
## Pantalla de inicio

Cada uno de los link que están en esta pantalla indican nos dan información sobre el H. Consejo Provincial de Chimborazo.

## MENÚ REPORTES

El link de Consulta Analfabetos nos indica que podemos ver los reportes tanto gráfico como estadístico los mismos que pueden ser visualizados por los diferentes usuarios que visiten nuestro sitio, como se muestra a continuación.

## Reporte Grafico



## Reporte Estadístico

Cantón	Habitantes (Hombres/Mujeres)	Total Habitantes	Analfabetos (Hombres/Mujeres)	Total Analfabetos
Alajaló	12 / 12	24	12 / 12	24
Chumbo	2500 / 1800	4300	294 / 294	588
Chunchi	1800 / 1500	3300	552 / 560	1112
Cacha	8800 / 8500	17300	514 / 578	1092
Canabari	3500 / 1800	5300	347 / 301	648
Guano	3855 / 5581	9437	3458 / 3032	6490
Palenque	12451 / 10258	22709	1021 / 1423	2444
Rosamba	3153 / 2452	5605	801 / 595	1396
Piapi	2451 / 2551	5002	257 / 274	531
Total	110844 / 98476	209320	13306 / 14604	27910

Una vez ingresado al sitio se mostrará al lado derecho del mismo el inicio de sesión en el cual deberá ingresar el nombre y su contraseña.

Personal

Nombre:

Contraseña:

Aceptar

**Nota:** El sistema dispone de dos tipos de usuarios que son el Administrador y el Limitado, los cuales pueden acceder a todo el sistema con la diferencia que el Administrador tiene el control absoluto sobre los usuarios que manejarán el sistema y el Administrador no lo puede.

H. CONSEJO PROVINCIAL DE CHIMBORAZO

OMEGA SISTEMA PARA EL CONTROL DEL PROGRAMA DE ALFABETIZACIÓN "YO SI PUEDO"

Administrador Personal Rep. Gráficos Rep. Estadísticos Cerrar Sesión

Bienvenido a SISALFA

Programa de Alfabetización

**Incorporación de Nuevos Alfabetizados**

El Programa de Alfabetización "Yo Si Puedo" promovido por el H. Consejo Provincial de Chimborazo, el Instituto Pedagógico Latinoamericano y Caribeño, con el apoyo de los Municipios y Directores Provinciales de Educación Hispana y Bilingüe, incorporará a 893 mujeres y 104 hombres de poblaciones rurales pertenecientes a los cantones Canabari, Guano, Palenque y Rosamba. Un total de 997 nuevos alfabetizados.

**Convenio para la Ejecución del Programa**

Cumplido con el mandato del pueblo chimboracense, el Consejo Provincial a través del Departamento de Gestión Social promueve el Programa de Alfabetización "Yo si puedo". Para constatación del convenio de cooperación firman el compromiso el Sr. Prefecto y una representante de una comunidad.

## Menú Administrador



En este menú podemos administrar la información referente a los estudiantes inscritos en el Programa, los estudiantes que se matriculan en los diferentes niveles, los materiales asignados a cada cantón para su mejor desempeño de la educación, los periodos académicos y los usuarios que tendrán acceso al sistema

## Menú Personal



En este menú podemos administrar la información referente a los coordinadores, asesores y educadores que están a cargo del Programa de Alfabetización.

## Menú Ubicación



En este menú podemos administrar la información referente a los cantones, parroquias y comunidades referente a la población y el número de analfabetismo que existe.

## Menú Reportes



En este menú podemos obtener los reporte de los analfabetos, estudiantes matriculados, estudiantes graduados y gasto del material asignado a cada cantón.































## Ingreso, Actualización, Eliminar y Listar Datos





En cada uno de los menús se podrá observar esta imagen la cual indica solo el significado de cada imagen (Nuevo, Actualiza, Eliminar, Listar) por lo cual no tiene resultado dar clic en esta imagen, para su respectiva manipulación de ingreso de datos en el sistema.



Estos botones estarán en la parte derecha de cada fila el listado de de datos de cada submenú, para lo cual si queremos hacer una actualización, eliminación, visualización, debemos dar clic en la imagen deseada.

### Lista de Datos

Cantón	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripción	Acción
	Hombres	Mujeres		Hombres	Mujeres			
Alausi	12	12	24	12	12	24	Canton	  
Chambo	2500	1800	4300	594	594	1188	Canton	  
Chunchi	1800	1500	3300	552	560	1112	Canton	  
Colta	6800	6500	13300	514	578	1092	Canton	  
Cumandá	3500	1800	5300	347	301	648	Canton	  
Guamote	9856	6581	16437	3458	3012	6470	Canton	  
Guano	12451	10256	22707	1521	1423	2944	Canton	  
Pallatanga	3126	3452	6578	601	100	701	Canton	  
Penipe	2451	2561	5012	257	274	531	Canton	  
Riobamba	65548	64014	129562	7500	7800	15300	Canton	  

 |  |  | 

### Ingreso de Datos

#### Registro de Datos

##### Datos del Cantón

Nombre Cantón:

Habitantes Hombres:

Habitantes Mujeres:

Analfabetos Hombres:

Analfabetos Mujeres:

Descripción:

## Actualización de Datos

Actualización de Datos

**Datos del Cantón:**

Cantón	Habitantes		Total Habitantes	Analfabetos		Total Analfabetos	Descripción
	Hombres	Mujeres		Hombres	Mujeres		
Alausi	12	12	24	12	12	24	Cantón

**Actualización de Datos:**

Nombre del Cantón:

Habitantes Hombres:

Habitantes Mujeres:

Analfabetos Hombres:

Analfabetos Mujeres:

Descripción:

## Reporte Grafico de Estudiantes Matriculados



## Reporte Estadístico Estudiantes Matriculados

Lista de Matriculados

Cantón	Parroquia	Comunidad	Nombre	Apellido	Etap	Nivel	Matricula	Edad	Estado Civil	Sexo	F. Matricula
Alausi	San Juan	Llanoloma	gina	vallejo	Enero 08 - Febrero 08	Segundo	Ordinaria	12	Soltero	Hombre	12-12-2008
Alausi	San Juan	Chipcha	Ivan	BuDay	Abril 05 - Mayo 06	Segundo	Ordinaria	23	Soltero	Masculino	12/12/122
Alausi	San Juan	La Dolorosa	Marcelo juanito	Corral saul	Enero 07 - Marzo 05	Primero	Extraordinaria	23	Soltero	Masculino	12/12/2008
Alausi	San Juan	Llanoloma	pedro	anibal	Enero 07 - Marzo 07	Primero	Extraordinaria	66	Casado	Hombre	12/12/2007
Chambo	Tixannn	La Silveria	María	Fernandez	Enero 07 - Marzo 05	Quinto	Extraordinaria	25	Soltero	Masculino	12/12/2008
Guano	Cacha	Lima Pamba	Carlos	Chavez	Abril 05 - Mayo 06	Tercero	Ordinaria	21	Divorciado	Masculino	12/12/2008

## Cerrar Sesión

Si damos clic en este botón, automáticamente saldremos a la página principal.

